B 1

```
UPDPSN - UPDATE PSN IN CETAB    MACRO V05.03b Saturday 29-Jun-85 05:55  Page 6-1
$UPPSN - UPDATE PSN DEFINITION

    218 000364  000207                      RETURN
    219                               ;
    220                               ; ERROR CONDITIONS
    221                               ;
    222 000366  000167  000000G       101$:   JMP     .CONER          ; CONSISTENCY ERROR
    223
    224         000001                          .END
```

```
UPD

SYM

SYM

CS.
F.N
ONS
PVC
PVC
PV$
PV$
PV$
P$V
P$V
P$V
P$V
P$V
P$V
P$V
P$V
SET
$BU
$MV
$OE
$OF
$PT
$PT
$PT
$PT
$UF
$WD
.BU
.CC
.FL
.TE
```

C 1

```
UPDPSN - UPDATE PSN IN CETAB    MACRO V05.03b Saturday 29-Jun-85 05:55  Page 6-2
Symbol table
```

```
UPD
```

```
                                                                                B  2
UPDPVC      CREATED BY  MACRO  ON 29-JUN-85 AT 05:55        PAGE 1

SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL  VALUE              REFERENCES

CS.FLG  = ******   GX       6-184
F.NRBD  = ******   GX      *6-157        *6-187        *6-189
ONSTR     000010 R         #5-137         7-209
PVCDF     000000 R         #5-134         5-135         6-163
PVCLN   = 000010           #5-135         6-164
PV$OWN  = ******   GX       6-183
PV$PSZ  = ******   GX       6-181
PV$WSZ  = ******   GX       6-182
P$VCT   = ******   GX       6-179
P$VFLG  = ******   GX       6-184
P$VLCN  = ******   GX       6-178
P$VNAM  = ******   GX       6-170         6-173         6-174         6-177
P$VOWN  = ******   GX       6-167         6-183
P$VPSZ  = ******   GX       6-181
P$VSTA  = ******   GX       7-206
P$VWSZ  = ******   GX       6-182
SETSTA    000402 R          6-180        #7-203
$BUFI   = ******   GX       6-189
$MVASC  = ******   GX       6-165
$OBUF   = ******   GX      *6-166         6-171        *6-172        *6-175         6-185
$OFDB   = ******   GX      *6-157        *6-187        *6-189
$PTNMB  = ******   GX       6-173         6-174         6-184
$PTNUM  = ******   GX       6-178         6-179         6-181         6-182
$PTOPT  = ******   GX       6-181         6-182         6-183         6-184
$PTR50  = ******   GX       6-170         6-177         6-183         7-211
$UPPVC    000014 RG        #6-156
$WDATA  = ******   GX       6-188
.BUFF   = ******   GX       6-157         6-162         6-186
.COMMA  = ******   GX      *6-161        *6-169
.FLAGS  = ******   GX      *6-160
.TEMP   = ******   GX      *6-158         7-203        *7-209        *7-212
```

```
                                                                                C  2
UPDPVC      CREATED BY  MACRO  ON 29-JUN-85 AT 05:55        PAGE 2
```

Table of contents

```
     6-  151     $UPREM - SETUP REMOTE DEFINITION
     7-  202     SETNAM - WRITE OUT REMOTE NODE NAME
     8-  228     SETARE - WRITE OUT AREA
     9-  254     SETADD - WRITE OUT REMOTE NODE ADDRESS
    10-  277     UPDLL  - UPDATE DOWNLINE LOAD DATABASE
```

```
UPDREM     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55      PAGE 3

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

DBGTP$           #5-120
ISTAT$           #4-68        5-120
MTRAN$           #5-120
PRINT            #4-59
STATE$           #4-68        5-123       #5-126      #5-129      #5-132      #5-135      #5-138      #5-141      #5-146      #5-149
TRAN$            #4-68       #5-124       #5-127      #5-130      #5-133      #5-136      #5-139      #5-142      #5-147
```

```
**FILE**ID**UPDROU
```

Table of contents

```
     4-   47     MACRO CALLS AND LOCAL DATA
     5-   93     $UPSER - SETUP REMOTE DEFINITION
     6-  157     PUTNOD - WRITE OUT NODE ADDRESS
     7-  194     PUTR50 - OUTPUT EXTENDED RAD50 STRING
     8-  222     PUTOCT  - OUTPUT AN OCTAL NUMBER
     9-  249     PUTFLG - OUTPUT FLAGS
```

Table of contents

```
     6-  219     $UPSLT - UPDATE SLT DEFINITION
     7-  261     SETNAM - CHECK DDM PROCESS NAME
     8-  287     SETFLG - SETUP THE FLAGS WORD
     9-  351     SETCNT - CHECK THE CONTROLLER NUMBER
    10-  371     SETUNT - CHECK THE UNIT NUMBER
    10-  392     SETCTT - SET OPTIONAL COUNTER TIMER VALUE
    10-  408     SETHTM - SET OPTIONAL HELLO TIMER VALUE
    10-  409     SETPPR - SET OPTIONAL ROUTER PRIORITY
    10-  410     SETBRA - SET BROADCAST ROUTER ADJACENCIES
    11-  439     SETTYP  - SET LINE TYPE
    439                                                  .SBTTL  SETTYP  - SET LINE TYPE
    440                                          ;+
    441                                          ; ** SETTYP - SET LINE TYPE
    442                                          ;
    443                                          ; INPUT          .TEMP = ADDRESS OF TEMPLATE
    444                                          ;
    445                                          ; OUTPUT         IF PRESENT, TYPE FIELD UPDATED.
    446                                          ;-
    447
    448 000774                                   SETTYP:
    449 000774  016701  000000G                          MOV     .TEMP,R1            ; POINT TO TEMPLATE
    450 001000  032761  000000G 000000G                  BIT     #SL$TYP,C.STS(R1)  ; TYPE SPECIFIED ?
    451 001006  001420                                   BEQ     20$                 ; IF EQ, NO
    452 001010  010146                                   MOV     R1,-(SP)            ; SAVE TEMPLATE ADDRESS
    453 001012  012702  000070'                          MOV     #MASTER,R2          ; ASSUME TYPE = MASTER
    454 001016  105761  000000G                          TSTB    S$LTYP(R1)          ; CHECK TYPE
    455 001022  001001                                   BNE     10$                 ; IF NE, MASTER
    456 001024  022222                                   CMP     (R2)+,(R2)+         ; ELSE TYPE = SLAVE
    457 001026  005067  000000G                  10$:    CLR     .TEMP               ; POINT TO START OF STRINGS
    458 001032  012701  000002                           MOV     #2,R1               ; 2 RAD50 WORDS
    459 001036  004767  000000G                          CALL    $PTR50              ; OUTPUT RAD50
    460 001042  012667  000000G                          MOV     (SP)+,.TEMP         ; RESTORE TEMPLATE ADDRESS
    461 001046  000207                                   RETURN
    462
    463 001050  005267  000000G                  20$:    INC     .COMMA              ; OPTION NOT SELECTED
    464 001054  000207                                   RETURN
    465
    466         000001                                   .END
```

```
                                        .SBTTL  SETAPR - SET THE STATION ACTIVE POLLING RATIO
                                ;+
                                ; *** - SETAPR - SET THE STATION ACTIVE POLLING RATIO
                                ;
                                ; INPUTS:
                                ;       TEMP = TEMPLATE ADDRESS
                                ;       $OBUF = POINTER INTO OUTPUT BUFFER
                                ;
                                ; OUTPUTS:
                                ;       THE APR IS WRITTEN TO THE OUTPUT BUFFER
                                ;
                                ;
                                ;-

000226                          SETAPR:
000226  016702  177546                  MOV     TEMP,R2         ; GET THE TEMPLATE ADDRESS
000232  116201  000010                  MOVB    C.APR(R2),R1    ; GET THE STATION APR
000236  016700  000000G                 MOV     $OBUF,R0        ; GET THE CURRENT POSITION IN THE OUTPUT BUFFER
000242  004767  000000G                 CALL    $CNVO           ; CONVERT THE NUMBER TO DECIMAL ASCII
000246  010067  000000G                 MOV     R0,$OBUF        ; UPDATE POINTER INTO OUTPUT BUFFER
000252  000207                          RETURN
```

```
    144                                         .SBTTL  $UPSVC - UPDATE SWITCHED VIRTUAL CIRCUIT
    145                                 ;+
    146                                 ; *** - $UPSVC - UPDATE SWITCHED VIRTUAL CIRCUIT
    147                                 ;
    148                                 ; INPUT:
    149                                 ;       R5 - TEMPLATE ADDRESS
    150                                 ;
    151                                 ; OUTPUT:
    152                                 ;       C-BIT = SUCCESS/FAILURE
    153                                 ;       $SVUPD SET TO INDICATE UPDATE ALREADY DONE
    154                                 ;       ALL OF THE SVC$DF TEMPLATES ARE OUTPUT.
    155                                 ;
    156                                 ;-
    157
    158 000022                          $UPSVC::
    159 000022  010546                          MOV     R5,-(SP)                ; SAVE R5
    160 000024  005767  000000G                 TST     $SVUPD                  ; UPDATED YET
    161 000030  001152                          BNE     20$                     ; IF NE, YES - ALL DONE
    162 000032  005267  000000G                 INC     $SVUPD                  ; MARK AS DONE
    163 000036  012705  000000G                 MOV     #$TMLST,R5              ; ELSE POINT OT LISTHEAD
    164 000042  011505                  10$:    MOV     (R5),R5                 ; NEXT TEMPLATE
    165 000044  001544                          BEQ     20$                     ; BR IF END OF LIST
    166 000046  122765  000000G 000000G         CMPB    #CS.SVC,C.STS(R5)       ; CHECK TEMPLATE TYPE
    167 000054  001372                          BNE     10$                     ; BR IF NOT SVC$DF
    168 000056  012767  000000G 000000C         MOV     #.BUFF,$OFDB+F.NRBD+2   ; SET NEW OUTPUT BUFFER
    169 000064  010567  000000G                 MOV     R5,.TEMP                ; SAVE TEMPLATE ADDRESS
    170 000070  005725                          TST     (R5)+                   ; POINT TO C.STS
    171 000072  010567  000000G                 MOV     R5,.FLAGS               ; SAVE ADDRESS OF FLAGS WORD
    172 000076  012767  177776  000000G         MOV     #-2,.COMMA              ; SUPPRESS FIRST 2 COMMAS
    173 000104  012700  000000G                 MOV     #.BUFF,R0               ; POINT AT START OF OUTPUT BUFFER
    174 000110  012701  000000'                 MOV     #SVCDF,R1               ; POINT AT START OF SVC DEFINITION
    175 000114  012702  000014                  MOV     #SVCLN,R2               ; GET LENGTH OF SVC DEFINITION
    176 000120  004767  000000G                 CALL    $MVASC                  ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
    177 000124  010067  000000G                 MOV     R0,$OBUF                ; SAVE CURRENT POSITION IN OUTPUT BUFFER
    178 000130                                  $PTNUM  #S$VNAM                 ; NAME (XX OF DLM-XX.YY)
    179 000142                                  $PTNUM  #S$VNAM+2               ; NAME (YY OF DLM-XX.YY)
    180 000154  005367  000000G                 DEC     $OBUF                   ; BACK OVER THE DECIMAL POINT
    181 000160                                  $PTNUM  #S$VPSZ,,#SV$PSZ        ; PACKET SIZE
    182 000204                                  $PTNUM  #S$VWSZ,,#SV$WSZ        ; WINDOW SIZE
    183 000230                                  $PTNUM  #S$VFLG,OCTAL           ; FLAGS
    184 000242                                  $PTNUM  #S$VRCL                 ; MAXIMUM RECALLS
    185 000254                                  $PTNUM  #S$VTIM                 ; RECALL TIMER
    186 000266                                  $PTR50  #S$VDTE,#5,#SV$DTE      ; RAD50 DTE NUMBER
    187 000314                                  $PTR50  #S$VOWN,#1              ; RAD50 OWNER OF SVC
    188 000330  016701  000000G                 MOV     $OBUF,R1                ; CALCULATE LENGTH OF BUFFER
    189 000334  162701  000000G                 SUB     #.BUFF,R1               ; ...
    190 000340  010167  000000C                 MOV     R1,$OFDB+F.NRBD         ; SET LENGTH OF BUFFER
    191 000344  004767  000000G                 CALL    $WDATA                  ; WRITE THE RECORD TO THE OUTPUT FILE
    192 000350  016705  000000G                 MOV     .TEMP,R5                ; POINT TO TEMPLATE
    193 000354  000632                          BR      10$                     ; CONTINUE
    194
    195 000356  012767  000000G 000000C 20$:    MOV     #$BUFI,$OFDB+F.NRBD+2   ; SET TO OUTPUT THE "END$DF" LINE
    196 000364  012605                          MOV     (SP)+,R5                ; RESTORE R5
    197 000366  000207                          RETURN
    198
    199         000001                          .END
```

```
                                                  .SBTTL  SETCW1 - SET THE CHARACTERISTICS WORD 1
                                          ;+
                                          ; *** - SETCW1 - SET THE CONTROLLER CHARACTERISTICS WORD 1
                                          ;
                                          ;
                                          ; INPUT:
                                          ;       .PNUMB = CHARACTERISTICS WORD 1
                                          ;       .PNUMH = CHARACTERISTICS WORD 1 (HIGH ORDER)
                                          ;       .TEMP = TEMPLATE ADDRESS
                                          ;
                                          ; OUTPUT:
                                          ;
                                          ;-

 000164                                   SETCW1:
 000164  016702  000000G                          MOV     .TEMP,R2                ; GET START OF TEMPLATE
 000170  016201  000010                           MOV     C.CW1(R2),R1    ; GET CHARACTERISTICS WORD 1
 000174  016700  000000G                          MOV     $OBUF,R0        ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
 000200  004767  000000G                          CALL    $CNVO           ; CONVERT NUMBER TO OCTAL
 000204  010067  000000G                          MOV     R0,$OBUF        ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
 000210  000207                                   RETURN
```

```
    114                                         ;
    115                                                 .MACRO  $PTR50  OFS,WC
    116                                                 MOV     OFS,R2
    117                                                 MOV     WC,R1
    118                                                 CALL    $PTR50
    119                                                 .ENDM
    120
    121                                                 .SBTTL  LOCAL DATA
    122
    123                                         ;
    124                                         ; DEFINE THE X.25 LEVEL 2 PARAMETER BLOCK
    125                                         ;
    126 000000                                          .ASECT
    127         000000                                  .=0
    128 000000                                          .BLKW   1               ; C.LNK
    129 000002                                          .BLKW   1               ; C.STS
    130 000004                                  C.LIN:  .BLKW   1               ; LINE ID: RAD50 PROCESS NAME,
    131 000006                                          .BLKW   2               ;          CONTROLLER, UNIT NUMBERS
    132 000012                                  C.WSZ:  .BLKW   1               ; WINDOW SIZE (0-7)
    133 000014                                  C.BSZ:  .BLKW   1               ; BLOCK SIZE
    134 000016                                  C.RTC:  .BLKW   1               ; RETRANSMIT COUNT
    135 000020                                  C.RTT:  .BLKW   1               ; RETRANSMIT TIMER VALUE
    136 000022                                  C.HBT:  .BLKW   1               ; HOLDBACK TIMER VALUE
    137         000024                          C.LEN=.                         ; LENGTH OF THE BLOCK
    138 000000                                          .PSECT
    139                                                 .NLIST  BEX
    140 000000     011      130      062        X2PDF:  .ASCII  <11>/X2P$DF/<11>
    141         000010                                  X2PLN=.-X2PDF
    142                                                 .LIST   BEX
    143
```

```
UPDX29 - UPDATE X.29 3 CHARACTE MACRO V05.03b Saturday 29-Jun-85 05:59  Page 6-1
Symbol table

F.NRBD= ****** GX       X29LN = 000010          $OBUF = ****** GX       $UPX29  000010RG        .COMMA= ****** GX
X$9CT = ****** GX       $BUFI = ****** GX       $OFDB = ****** GX       $WDATA= ****** GX       .FLAGS= ****** GX
X$9MXC= ****** GX       $MVASC= ****** GX       $PTNUM= ****** GX       .BUFF = ****** GX       .TEMP = ****** GX
X29DF   000000R

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000136    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8783 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:04.59
SY:UPDX29.V2,[132,134]UPDX29/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDX29
```

```
UPDX29     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59     PAGE 1
```

```
**FILE**ID**MLD




MM      MM   LL           DDDDDDDD
MM      MM   LL           DDDDDDDD
MMMM  MMMM   LL           DD      DD
MMMM  MMMM   LL           DD      DD
MM  MM  MM   LL           DD      DD
MM  MM  MM   LL           DD      DD
MM      MM   LL           DD      DD
MM      MM   LL           DD      DD
MM      MM   LL           DD      DD
MM      MM   LL           DD      DD
MM      MM   LL           DD      DD        ....
MM      MM   LL           DD      DD        ....
MM      MM   LLLLLLLLLL   DDDDDDDD          ....
MM      MM   LLLLLLLLLL   DDDDDDDD          ....




LL               SSSSSSSS   TTTTTTTTTT
LL               SSSSSSSS   TTTTTTTTTT
LL             SS               TT
LL             SS               TT
LL             SS               TT
LL             SS               TT
LL               SSSSSS         TT
LL               SSSSSS         TT
LL                     SS       TT
LL                     SS       TT
LL                     SS       TT
LL                     SS       TT
LLLLLLLLLL     SSSSSSSS         TT
LLLLLLLLLL     SSSSSSSS         TT
```

```
MLD     - Microcode Loader      MACRO V05.03b Wednesday 28-Aug-85 11:26
Table of contents
```

```
 000666  016704  001370'                 MOV     LDBADR,R4               ;; Get APR6-biased LDB address
 000672                                  CALL    @BLXIO                  ;; Copy microcode to LDB
 000676  016704  001364'                 MOV     LDBPTR,R4               ;; Get address of LDB's CCB
 000702  010564  000020                  MOV     R5,C.CNT1(R4)           ;; Copy LDB byte count to CCB
 000706  016764  001370' 000016          MOV     LDBADR,C.BUF1+2(R4)     ;; Recopy address if driver changed it

                                         .ENDC                           ;; End UMR conditional


                                 ;       Pick up the information needed to dispatch to the driver and perform
                                 ;       the dispatch.  Return to user state after the dispatch - all registers
                                 ;         and mappings prior to the switch into system state are preserved across
                                 ;       the transition.

 000714  016702  001400'         10$:    MOV     PDVIDX,R2               ;; Get driver's PDV index
 000720  016705  001404'                 MOV     DDMTBL,R5               ;; Get line table address as well
 000724  010403                          MOV     R4,R3                   ;; Copy CCB address
 000726  062703  000010                  ADD     #C.FNC,R3               ;; Compute address of function code
 000732                                  CALLR   @PDSPL                  ;; Dispatch to driver at process level
                                                                         ;;    and return to user state at "20$"

 000736  105764  000011          20$:    TSTB    C.MOD(R4)               ; Clear C-bit and test for halt
 000742  001426                          BEQ     40$                     ; Branch if "halt" CCB, ignore status

                                         .IF     DF      N$$UMR          ; If system has UMRs ...

 000744  016705  001364'                 MOV     LDBPTR,R5               ; Get address of LDB's CCB
 000750  016503  000012                  MOV     C.STS(R5),R3            ; Get driver's load status
 000754  001011                          BNE     25$                     ; Branch if error(s) occurred
 000756  067764  177046  000016          ADD     @RDBSZ,C.BUF1+2(R4)     ; Advance load buffer address
 000764  167764  177040  000020          SUB     @RDBSZ,C.CNT1(R4)       ; Subtract enough data to fill an LDB
 000772  003304                          BGT     DSPDRV                  ; Branch if more of buffer to load
 000774  000241                          CLC                             ; Else clear C-bit to continue load
 000776                                  RETURN                          ; Exit

 001000                          25$:    .IFF                            ; If system has no UMRs ...

                                         MOV     C.STS(R4),R3            ; Else, get driver's completion status
                                         BEQ     40$                     ; Branch if loading is to continue

                                         .ENDC                           ; End UMR conditional

 001000  032703  077777                  BIT     #^C<CS.ERR>,R3          ; Errors other than logical data end?
 001004  001404                          BEQ     30$                     ; Branch if not
 001006  012702  000051'                 MOV     #DEVERR,R2              ; Get "device error" string address
 001012                                  CALL    LOGEVT                  ; Go log the event

 001016  000261                  30$:    SEC                             ; Set C-bit to indicate load halt
 001020                          40$:    RETURN                          ; Exit
```

```
FC.MLD=  000026        FS.STR=  001000        F.STBK=  000036        L$$ASG=  000000        N.ANM2=  000024
FC.PCT=  000030        FS.TRM=  003000        F.UNIT=  000136        L$$DRV=  000000        N.DID =  000024
FC.PWR=  000022        FS.WLB=  001000        F.URBD=  000020        L$$P11=  000001        N.DVNM=  000032
FC.RCE=  000002        FS.XKL=  002000        F.VBN =  000064        L$$11R=  000000        N.FID =  000000
FC.RCP=  000014        FS.XOF=  010000        F.VBSZ=  000060        L.COST   000015        N.FNAM=  000006
FC.TIM=  000010        FS.XON=  007000        G$$TPP=  000000        L.CTL    000012        N.FTYP=  000014
FC.XCP=  000012        FS.ZER=  002000        G$$TSS=  000000        L.CVA    177776        N.FVER=  000016
FC.XME=  000000        FX.DIR=  000004        G$$TTK=  000000        L.DDM    000002        N.NEXT=  000022
FDB      000144R   002 FX.DLN=  000003        G$$WRD=  000000        L.DDS    000004        N.STAT=  000020
FD.RWM=  ****** GX     FX.DMX=  000002        IE.EOF=  ****** GX     L.DLC    000003        N.UNIT=  000034
FILDEV   001326R   002 FX.LEN=  000000        IO.WLB=  ****** GX     L.DLM    000006        OP$INI=  000000
FILDIR   001334R   002 F$$LVL=  000001        ISAR5    000002RG  004 L.DLS    000010        OP$TER=  000001
FILERR   001310R   002 F.ACTL=  000076        I$$RAR=  000000        L.FLG    000000        PARHD    000010RG  004
FIOERR   000073R   002 F.ALOC=  000040        I$$RDN=  000000        L.KRBA   000016        PAR$$$=  000027
FO.RD =  ****** GX     F.BBFS=  000062        KISAR5=  ****** GX     L.LEN =  000022        PDSPL    000024RG  004
FR$BCC=  000007        F.BDB =  000070        K$$CNT=  177546        L.MPF    000022        PDVIDX   001400R   002
FR$CCF=  000001        F.BGBC=  000057        K$$CSR=  177546        L.NMST   000020        PDVTA    000026RG  004
FR$CDF=  000002        F.BKDN=  000026        K$$LDC=  000000        L.NSTA   000014        PH$HDE=  000004
FR$DAO=  000011        F.BKDS=  000020        K$$TPS=  000074        L.OWNR   000021        PH$LOC=  000002
FR$EXC=  000000        F.BKEF=  000050        LDBADR   001370R   002 L.UNT    000013        PH$MTS=  000003
FR$FRM=  000010        F.BKP1=  000051        LDBGT    000020RG  004 MLD      000036R   004 PH$UMP=  000000
FR$FTL=  000005        F.BKST=  000024        LDBIAS   001366R   002 MLDERR   000000R   002 PH$WCS=  000001
FR$OPN=  000004        F.BKVB=  000064        LDBPTR   001364R   002 MLOAD    001506R   004 PWRAST   001152R   004
FR$RFD=  000006        F.CHR =  000075        LDBRT    000022RG  004 MO$SAC=  000016        P$$P45=  000000
FR$SBU=  000012        F.CNTG=  000034        LDBWTS=  000001        MO$SPR=  000012        P$$WRD=  000000
FR$SHO=  000003        F.DFNB=  000046        LD$LP =  000000        MO$SSV=  000014        QRMVF    000012RG  004
FR$UBU=  000013        F.DSPT=  000044        LF.ACT=  100000        MO$25A=  000006        Q$$OPT=  000010
FR$UPT=  000014        F.DVNM=  000134        LF.BRO=  000400        MO$25P=  000002        RDBSZ    000030RG  004
FS.AST=  000000        F.EFBK=  000010        LF.BWT=  000007        MO$25S=  000004        RE$ADC=  000004
FS.CIB=  002000        F.EFN =  000050        LF.ENA=  002000        MO$29S=  000010        RE$ADF=  000017
FS.CRA=  001000        F.EOBB=  000032        LF.LPB=  001000        MXFNLN=  000006        RE$ADR=  000007
FS.DIS=  013000        F.ERR =  000052        LF.MDC=  000100        M$HIGH=  000003        RE$BLK=  000010
FS.DVC=  001000        F.EXT =  000072        LF.MFL=  004000        M$$CRB=  000124        RE$CAF=  000014
FS.ENB=  012000        F.FACC=  000043        LF.MTP=  000020        M$$CRX=  000000        RE$DAT=  000001
FS.EXI=  001000        F.FFBY=  000014        LF.PAC=  000200        M$$FCS=  000000        RE$DRP=  000016
FS.GET=  006000        F.FLG =  000074        LF.RDY=  040000        M$$MGE=  000000        RE$[illegible]DT=  000013
FS.HLT=  000000        F.FNAM=  000110        LF.REA=  010000        M$$NET=  000000        RE$[illegible]SN=  000012
FS.INI=  000000        F.FNB =  000102        LF.SER=  000040        M$$OVR=  000000        RE$[illegible] =  000001
FS.KIL=  000000        F.FTYP=  000116        LF.TIM=  000010        M$3100=  000000        RE$[illegible]=  000004
FS.LCL=  100000        F.FVER=  000120        LF.UNL=  020000        M$3101=  000001        RE$[illegible]=  000000
FS.LTM=  001000        F.HIBK=  000004        LF.X2P=  000000        M$3102=  000002        RE$RCV=  000001
FS.MNT=  004000        F.LUN =  000042        LN.CLO=  000000        M$3103=  000003        RE$SED=  000011
FS.MSN=  014000        F.MBCT=  000054        LN.DUM=  000005        M.KTAE=  000010        RE$SKW=  000006
FS.REA=  001000        F.MBC1=  000055        LN.LOA=  000004        M.KTEF=  000002        RE$STA=  000002
FS.RET=  000000        F.MBFG=  000056        LN.LOO=  000003        M.KTMG=  000004        RE$SUM=  000003
FS.REZ=  003000        F.NRBD=  000024        LN.OAU=  000003        M.KTUN=  000006        RE$SYN=  000000
FS.RLB=  002000        F.NREC=  000030        LN.OFF=  000001        NTUIC    000006RG  004 RE$TMF=  000021
FS.RNG=  011000        F.OVBS=  000030        LN.ON =  000000        N$$ACC=  000001        RE$TMO=  000000
FS.RST=  000000        F.RACC=  000016        LN.OOP=  000004        N$$BUF=  000001        RE$TMR=  000020
FS.RTN=  001000        F.RATT=  000001        LN.OPE=  000001        N$$LDV=  000001        RE$UPT=  000002
FS.SET=  005000        F.RCNM=  000034        LN.REF=  000002        N$$MCP=  000001        RE$URE=  000003
FS.SFC=  005000        F.RCTL=  000017        LN.SER=  000002        N$$MLL=  000001        RE$VER=  000005
FS.SFR=  006000        F.RSIZ=  000002        LN.STA=  000017        N$$MOV=  000010        RE$VRQ=  000015
FS.SFS=  004000        F.RTYP=  000000        LN.SUB=  000360        N$$NCT=  000001        RT$INI=  000002
FS.SPW=  040000        F.SEQN=  000100        LN.TRI=  000006        N$$PEM=  000001        RT$OFF=  000001
FS.STM=  000000        F.SPDV=  000072        LOCFIL   001372R   004 N$$UMR=  000001        RT$ON =  000000
FS.STP=  002000        F.SPUN=  000074        LOGEVT   001022R   004 N.ANM1=  000002        R$$DER=  000000
```

```
MMM         MMM     LLL                 DDDDDDDDDDDD              111                 111              SSSSSSSSSSSS
MMM         MMM     LLL                 DDDDDDDDDDDD              111                 111              SSSSSSSSSSSS
MMM         MMM     LLL                 DDDDDDDDDDDD              111                 111              SSSSSSSSSSSS
MMMMMM   MMMMMM     LLL                 DDD         DDD        111111              111111           SSS
MMMMMM   MMMMMM     LLL                 DDD         DDD        111111              111111           SSS
MMMMMM   MMMMMM     LLL                 DDD         DDD        111111              111111           SSS
MMM   MMM   MMM     LLL                 DDD         DDD           111                 111           SSS
MMM   MMM   MMM     LLL                 DDD         DDD           111                 111           SSS
MMM   MMM   MMM     LLL                 DDD         DDD           111                 111           SSS
MMM         MMM     LLL                 DDD         DDD           111                 111              SSSSSSSSS
MMM         MMM     LLL                 DDD         DDD           111                 111              SSSSSSSSS
MMM         MMM     LLL                 DDD         DDD           111                 111              SSSSSSSSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLL                 DDD         DDD           111                 111                       SSS
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD           111111111           111111111        SSSSSSSSSSSS
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD           111111111           111111111        SSSSSSSSSSSS
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD           111111111           111111111        SSSSSSSSSSSS
```

```
**FILE**ID**MLD11S
```

```
                                                          C  1
UPDPSN - UPDATE PSN IN CETAB    MACRO V05.03b Saturday 29-Jun-85 05:55  Page 6-2
Symbol table

CS.CHN= ****** GX       CS.TLO= ****** GX       DISTBL  000010R         $PTOPT= ****** GX       $UPRDT= ****** GX
CS.CUG= ****** GX       CS.X2P= ****** GX       F.NRBD= ****** GX       $PTR50= ****** GX       $UPX2P= ****** GX
CS.DSA= ****** GX       CS.X29= ****** GX       PSNDF   000000R         $UPCHN= ****** GX       $UPX29= ****** GX
CS.DSC= ****** GX       CS.X3P= ****** GX       PSNLN = 000010          $UPCUG= ****** GX       $UPX3P= ****** GX
CS.DST= ****** GX       C.FLG   000012          $BUFI = ****** GX       $UPDSA= ****** GX       $WDATA= ****** GX
CS.DTE= ****** GX       C.LEN = 000020          $MVASC= ****** GX       $UPDSC= ****** GX       .BUFF = ****** GX
CS.FLG= ****** GX       C.NAM   000004          $OBUF = ****** GX       $UPDST= ****** GX       .COMMA= ****** GX
CS.PSN= ****** GX       C.POR   000010          $OFDB = ****** GX       $UPDTE= ****** GX       .CONER= ****** GX
CS.PVC= ****** GX       C.STS = ****** GX       $PSIPT= ****** GX       $UPPSN  000072RG        .FLAGS= ****** GX
CS.RDT= ****** GX       C.THI   000016          $PTNUM= ****** GX       $UPPVC= ****** GX       .TEMP = ****** GX
CS.THI= ****** GX       C.TLO   000014

. ABS.  000020    000   (RW,I,GBL,ABS,OVR)
        000372    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8791 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.29
SY:UPDPSN.V2,[132,134]UPDPSN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDPSN
```

```
                                                          D  1
UPDPSN     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 1
```

```
UPDPVC     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 2

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME        REFERENCES

EPRINT            #4-79
ERROR$            #4-68
PRINT             #4-67
$PTNMB            #4-112      6-173      6-174      6-184
$PTNUM            #4-111      6-178      6-179      6-181      6-182
$PTR50            #4-116      6-170      6-177      6-183
$$  TN            #4-91       4-111      4-112
```

```
**FILE**ID**UPDRDT
```

```
                                .TITLE  UPDREM - CFE UPDATE REMOTE DEFINITION IN CETAB
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       STATE TABLE TO PARSE THE REMOTE DEFINITIONS
                        ;
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;
                        ;


```

```
**FILE**ID**UPDROU










UU      UU    PPPPPPPP      DDDDDDDD      RRRRRRRR        OOOOOO      UU      UU
UU      UU    PPPPPPPP      DDDDDDDD      RRRRRRRR        OOOOOO      UU      UU
UU      UU    PP      PP    DD      DD    RR      RR    OO      OO    UU      UU
UU      UU    PP      PP    DD      DD    RR      RR    OO      OO    UU      UU
UU      UU    PP      PP    DD      DD    RR      RR    OO      OO    UU      UU
UU      UU    PP      PP    DD      DD    RR      RR    OO      OO    UU      UU
UU      UU    PPPPPPPP      DD      DD    RRRRRRRR      OO      OO    UU      UU
UU      UU    PPPPPPPP      DD      DD    RRRRRRRR      OO      OO    UU      UU
UU      UU    PP            DD      DD    RR  RR        OO      OO    UU      UU
UU      UU    PP            DD      DD    RR  RR        OO      OO    UU      UU
UU      UU    PP            DD      DD    RR    RR      OO      OO    UU      UU       ....
UU      UU    PP            DD      DD    RR    RR      OO      OO    UU      UU       ....
UUUUUUUUUU    PP            DDDDDDDD      RR      RR      OOOOOO      UUUUUUUUUU       ....
UUUUUUUUUU    PP            DDDDDDDD      RR      RR      OOOOOO      UUUUUUUUUU       ....






LL                SSSSSSSS    TTTTTTTTTT
LL                SSSSSSSS    TTTTTTTTTT
LL            SS                TT
LL            SS                TT
LL            SS                TT
LL            SS                TT
LL              SSSSSS          TT
LL              SSSSSS          TT
LL                    SS        TT
LL                    SS        TT
LL                    SS        TT
LL                    SS        TT
LLLLLLLLLL    SSSSSSSS          TT
LLLLLLLLLL    SSSSSSSS          TT
```

```
        .TITLE  UPDSER - CFE UPDATE REMOTE CHARACTERISTICS IN CETAB
        .IDENT  /V05.00/
        .ENABL  LC
;
; COPYRIGHT (C) 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       STATE TABLE TO PARSE THE REMOTE DEFINITIONS
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
;
```

```
                                .TITLE  UPDSLT - CFE UPDATE SLT DEFINITION IN CETAB
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       STATE TABLE TO PARSE THE LINE DEFINITIONS
                        ;
                        ;
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;
```

```
                                                      C  7
UPDSLT - CFE UPDATE SLT DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:56  Page 11-1
Symbol table

BITS     000126R     002 LF.SER= ****** GX       SETFLG  000242R             S$LRPR= ****** GX       $PTNUM= ****** GX
CS.SLT= ****** GX        LF.TIM= ****** GX       SETHTM  000642R             S$LTYP= ****** GX       $PTOPT= ****** GX
C.CNT    000010          LF.X2P= ****** GX       SETNAM  000204R             $ALPHA= 000022          $PTR50= ****** GX
C.CTT    000012          LMDC    000007R         SETRPR  000670R             $ANY  = 000020          $RAD50= 000016
C.FLG    000006          LMDCL = 000007          SETTYP  000774R             $BLANK= 000006          $STRNG= 000004
C.LEN =  000014          LMFL    000025R         SETUNT  000566R             $CAT5 = ****** GX       $SUBXP= 000010
C.NAM    000004          LMFLL = 000007          SKIP    000112R         002 $CLNUP= ****** GX       $TMLST= ****** GX
C.STS =  ****** GX       LMTP    000016R         SLAVE   000074R             $DIGIT= 000024          $UPSLT  000100RG
C.UNT    000011          LMTPL = 000007          SLTKTB  000000RG        003 $DNUMB= 000014          $$$FLG= 177777
EXIT     000172R     002 LSER    000061R         SLTSTB  000000RG        002 $EOS  = 000012          $$$KEY= 000011
LBRO     000052R         LSERL = 000007          SL$BRA= ****** GX           $ERROR= ****** GX       $$$STA= 000000
LBROL =  000007          LTIM    000034R         SL$CTM= ****** GX           $EXIT = 000000          $$$TMP= 000043R      004
LENA     000000R         LTIML = 000007          SL$HTM= ****** GX           $FAIL = 177777          .COMMA= ****** GX
LENAL =  000007          LX2P    000043R         SL$RPR= ****** GX           $LAMDA= 000000          .CONER= ****** GX
LFS      000104R     002 LX2PL = 000007          SL$TYP= ****** GX           $MOVPT= ****** GX       .FLAGS= ****** GX
LF.BRO=  ****** GX       MASTER  000070R         S$LBRA= ****** GX           $MDVST= ****** GX       .PNUMB= ****** GX
LF.ENA=  ****** GX       NEXT    000122R     002 S$LCTT= ****** GX           $MVASC= ****** GX       .PSTPT= ****** GX
LF.MDC=  ****** GX       SETBRA  000732R         S$LFLG= ****** GX           $NUMBR= 000002          .TEMP = ****** GX
LF.MFL=  ****** GX       SETCNT  000540R         S$LHTM= ****** GX           $DBUF = ****** GX       .TPARS= ****** GX
LF.MTP=  ****** GX       SETCTT  000614R

. ABS.   000014     000   (RW,I,GBL,ABS,OVR)
         001056     001   (RW,I,LCL,REL,CON)
$STATE   000200     002   (RW,D,LCL,REL,CON)
$KTAB    000024     003   (RW,D,LCL,REL,CON)
$KSTR    000047     004   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10808 Words  ( 43 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:40.36
SY:UPDSLT.V2,[132,134]UPDSLT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDSLT
```

```
                                                       D  7
UPDSLT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 1
```

```
                                                .SBTTL  SETFLG - SET STATION FLAGS
                                        ;+
                                        ; *** - SETFLG - SET STATION FLAGS
                                        ;
                                        ; INPUTS:
                                        ;       TEMP = TEMPLATE ADDRESS
                                        ;       $OBUF = POINTER INTO OUTPUT BUFFER
                                        ;
                                        ; OUTPUTS:
                                        ;       THE FLAGS ARE WRITTEN TO THE OUTPUT BUFFER
                                        ;
                                        ;-

 000254                                 SETFLG:
 000254  010446                                 MOV     R4,-(SP)                ; SAVE R4
 000256  016704  177516                         MOV     TEMP,R4                 ; POINT AT TEMPLATE
 000262  016700  000000G                        MOV     $OBUF,R0                ; POINT AT OUTPUT BUFFER
 000266  032764  000000G 000006                 BIT     #SF.ENA,C.FLG(R4)       ; IS THE STATION TO BE ENABLED ?
 000274  001406                                 BEQ     10$                     ; IF EQ, NO
 000276  012701  000002'                        MOV     #SENA,R1                ; GET ADDRESS OF ENABLE STRING
 000302  012702  000007                         MOV     #SENAL,R2               ; AND LENGTH
 000306  004767  000000G                        CALL    $MVASC                  ; MOVE STRING INTO OUTPUT BUFFER

 000312  032764  000000G 000006         10$:    BIT     #SF.SVC,C.FLG(R4)       ; IS THE STATION AN SVC ?
 000320  001406                                 BEQ     20$                     ; IF EQ, NO
 000322  012701  000020'                        MOV     #SSVC,R1                ; GET ADDRESS OF STRING
 000326  012702  000007                         MOV     #SSVCL,R2               ; AND LENGTH
 000332  004767  000000G                        CALL    $MVASC                  ; MOVE STRING INTO OUTPUT BUFFER

 000336  032764  000000G 000006         20$:    BIT     #SF.MFL,C.FLG(R4)       ; IS THE STATION MARKED FOR LOAD ?
 000344  001406                                 BEQ     30$                     ; IF EQ, NO
 000346  012701  000011'                        MOV     #SMFL,R1                ; GET ADDRESS OF STRING
 000352  012702  000007                         MOV     #SMFLL,R2               ; AND LENGTH
 000356  004767  000000G                        CALL    $MVASC                  ; MOVE STRING INTO OUTPUT BUFFER

 000362  032764  000000G 000006         30$:    BIT     #SF.SER,C.FLG(R4)       ; SERVICE DISABLED ?
 000370  001406                                 BEQ     35$                     ; NO
 000372  012701  000027'                        MOV     #SSER,R1                ; GET ADDRESS
 000376  012702  000007                         MOV     #SSERL,R2               ; LENGTH
 000402  004767  000000G                        CALL    $MVASC                  ; MOVE IT

 000406  020067  000000G                35$:    CMP     R0,$OBUF                ; ANY STRINGS ADDED ?
 000412  001401                                 BEQ     40$                     ; IF EQ, NO
 000414  105740                                 TSTB    -(R0)                   ; ELSE, SKIP OVER LAST "!"
 000416  112720  000054                 40$:    MOVB    #<',>,(R0)+             ; INSERT DELIMETER
 000422  010067  000000G                        MOV     R0,$OBUF                ; UPDATE POINTER INTO OUTPUT BUFFER
 000426  012604                                 MOV     (SP)+,R4                ; RESTORE R4
 000430  000207                                 RETURN
```

```
UPDSVC - UPDATE SWITCHED VIRTUA MACRO V05.03b Saturday 29-Jun-85 05:58  Page 6-1
Symbol table

CS.SVC= ****** GX       SVCDF    000000R        S$VNAM= ****** GX       $MVASC= ****** GX       $TMLST= ****** GX
C.STS = ****** GX       SVCLN = 000014          S$VOWN= ****** GX       $OBUF  = ****** GX      $UPSVC   000022RG
F.NRBD= ****** GX       SV$DTE= ****** GX       S$VPSZ= ****** GX       $OFDB  = ****** GX      $WDATA= ****** GX
OFFLEN= 000003          SV$PSZ= ****** GX       S$VRCL= ****** GX       $PTNUM= ****** GX       .BUFF  = ****** GX
OFFSTA  000016R         SV$WSZ= ****** GX       S$VTIM= ****** GX       $P1OPT= ****** GX       .COMMA= ****** GX
ONLEN = 000002          S$VDTE= ****** GX       S$VWSZ= ****** GX       $PTR50= ****** GX       .FLAGS= ****** GX
ONSTA   000014R         S$VFLG= ****** GX       $BUFI  = ****** GX      $SVUPD= ****** GX       .TEMP  = ****** GX


. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000370    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8830 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.26
SY:UPDSVC.V2,[132,134]UPDSVC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDSVC
```

```
UPDSVC     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58      PAGE 1
```

```
                                      .SBTTL  SETCSR - SET THE SECONDARY CSR ADDRESS
                              ;+
                              ; *** - SETCSR - SET THE SECONDARY CSR ADDRESS
                              ;
                              ; INPUT:
                              ;       .TEMP = TEMPLATE ADDRESS
                              ;
                              ; OUTPUT:
                              ;
                              ;-

000212                        SETCSR: $PTNUM  #C.CSR,OCTAL,#UN$SCS
000236  000207                        RETURN
```

```
                                                .SBTTL  $UPX2P - UPDATE X.25 LEVEL 2 PROTOCOL DEFINITION
                                        ;+
                                        ;*** - $UPX2P - UPDATE X.25 LEVEL 2 PROTOCOL DEFINITION
                                        ;
                                        ; INPUT:
                                        ;       R3-R5   - TPARS REGISTERS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

 000010                                 $UPX2P::
 000010 012767  000000G 000000C                 MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
 000016 010567  000000G                         MOV     R5,.TEMP        ; SAVE TEMPLATE ADDRESS
 000022 005725                                  TST     (R5)+           ; POINT TO C.STS
 000024 010567  000000G                         MOV     R5,.FLAGS       ; SAVE ADDRESS OF FLAGS WORD
 000030 012767  177777  000000G                 MOV     #-1,.COMMA      ; SUPPRESS INITIAL COMMA
 000036 012700  000000G                         MOV     #.BUFF,R0       ; POINT AT START OF OUTPUT BUFFER
 000042 012701  000000'                         MOV     #X2PDF,R1       ; POINT AT START OF X2P DEFINITION
 000046 012702  000010                          MOV     #X2PLN,R2       ; GET LENGTH OF X2P DEFINITION
 000052 004767  000000G                         CALL    $MVASC          ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
 000056 010067  000000G                         MOV     R0,$OBUF        ; SAVE CURRENT POSITION IN OUTPUT BUFFER
 000062 004767  000112                          CALL    SETLIN          ; LINE ID
 000066                                         $PTNUM  #C.WSZ          ; WINDOW SIZE
 000100                                         $PTNUM  #C.BSZ          ; BLOCK SIZE
 000112                                         $PTNUM  #C.RTC          ; RETRANSMIT COUNT
 000124                                         $PTNUM  #C.RTT          ; RETRANSMIT TIMER
 000136                                         $PTNUM  #C.HBT          ; HOLDBACK TIMER
 000150 016701  000000G                         MOV     $OBUF,R1        ; CALCULATE LENGTH OF BUFFER
 000154 162701  000000G                         SUB     #.BUFF,R1       ;       ...
 000160 010167  000000C                         MOV     R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
 000164 004767  000000G                         CALL    $WDATA          ; WRITE THE RECORD TO THE OUTPUT FILE
 000170 012767  000000G 000000C                 MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
 000176 000207                                  RETURN
```

C 12

```
UPDX29     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE                REFERENCES

F.NRBD   =  ******   GX      *6-142       *6-156       *6-158
X$9CT    =  ******   GX       6-153
X$9MXC   =  ******   GX       6-152
X29DF       000000 R         #5-124        5-125        6-148
X29LN    =  000010           #5-125        6-149
$BUFI    =  ******   GX       6-158
$MVASC   =  ******   GX       6-150
$OBUF    =  ******   GX      *6-151        6-154
$OFDB    =  ******   GX      *6-142       *6-156       *6-158
$PTNUM   =  ******   GX       6-152        6-153
$UPX29      000010 RG        #6-141
$WDATA   =  ******   GX       6-157
.BUFF    =  ******   GX       6-142        6-147        6-155
.COMMA   =  ******   GX      *6-146
.FLAGS   =  ******   GX      *6-145
.TEMP    =  ******   GX      *6-143
```

D 12

```
UPDX29     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59      PAGE 2
```

Table of contents

```
     6-    49     Macro Definitions
     7-    93     Equated Symbols and Literals
     8-   161     Load Control Data
     9-   249     MLD       - Microcode Loader Main Routine
    10-   471     DSPDRV    - Dispatch to Device Driver
    11-   579     LOGEVT    - Log a Network Event
    12-   620     PWRAST    - Powerfail Recovery AST
    12-   621     ABOAST    - Abort Cleanup AST
    13-   691     FCSERR    - Input File FCS Error Handler
    14-   730     LOCFIL    - Locate Microcode File
    15-   772     MLOAD     - Load Control for RSX-11M Systems
```

```
                                        .SBTTL  LOGEVT  - Log a Network Event
                                ;+
                                ;                               *** LOGEVT ***
                                ;
                                ; Function:     To log microcode loading events to the network using the
                                ;               NM: network management pseudo-device.
                                ;
                                ; Call Format:  CALL    LOGEVT
                                ;
                                ; Inputs:       R2 = address of specific error ASCII string
                                ;               PDVIDX = PDV index for device driver
                                ;               CTLNUM = device controller number
                                ;
                                ; Outputs:      DECnet event is logged to the event sink for the node
                                ;               this program is running on.
                                ;
                                ; Registers:    R0, R1, R2 and R5 destroyed
                                ;
                                ;-

001022  010205                  LOGEVT: MOV     R2,R5                   ; Copy address of input string
001024  016746  001402'                 MOV     CTLNUM,-(SP)            ; Put controller number on stack
001030  017701  176772                  MOV     @PDVTA,R1               ; Get address of PDV table
001034  066701  001400'                 ADD     PDVIDX,R1               ; Advance to PDV address of driver
001040  011101                          MOV     (R1),R1                 ; Get driver's PDV address
001042  016146  000004                  MOV     Z.NAM(R1),-(SP)         ; Put RAD50 process name on stack

001046  012700  001546'                 MOV     #EVTBLK+E$DATA,R0       ; Get address of event output data
001052  012701  000000'                 MOV     #MLDERR,R1              ; Get $EDMSG edit instructions address
001056  010602                          MOV     SP,R2                   ; Get $EDMSG argument block address
001060                                  CALL    $EDMSG                  ; Form first ASCII string
001064  105020                          CLRB    (R0)+                   ; Make it an ASCIZ string
001066  022626                          CMP     (SP)+,(SP)+             ; Clean off the stack

001070  112520                  10$:    MOVB    (R5)+,(R0)+             ; Append a byte from input string
001072  001376                          BNE     10$                     ; End of string is marked by zero
001074  162700  001526'                 SUB     #EVTBLK,R0              ; Get event buffer size, log, and exit

001100                                  QIOW$S  #IO.WLB,#EVTLUN,#EVTEFL,,,,<#EVTBLK,R0,#ETYPCL,#0>
001150                                  RETURN
```

```
Symbol table

R$$K11= 000001         S2.BRQ= 000020         S6.EIO= 000400         T3.ACP= 100000         UU.RCT= 000002
R$$SND= 000000         S2.CR = 002000         S6.RDI= 100000         T3.CAL= 000100         UU.RDY= 000200
R$$11M= 000000         S2.CTO= 040000         S6.RLU= 001000         T3.CLI= 001000         UU.SER= 000001
SC$OFF= 000001         S2.CTS= 100000         TKTCB   000014RG   004 T3.GFL= 000010         UU.SIO= 001000
SC$ON = 000000         S2.ELF= 001000         TMREFL= 000003         T3.MCR= 004000         UU.SPC= 000040
SC$RST= 000003         S2.FLF= 000400         TS.BLK= 171700         T3.NET= 000020         U.ACB   000062
SC$SHU= 000002         S2.IRQ= 000200         TS.CKP= 000200         T3.NSD= 000200         U.ACP = 000032
SETTMR  001346R    002 S2.OBF= 004000         TS.CKR= 000100         T3.PMD= 040000         U.ADMA  000066
SF.ACT= 000200         S2.ORQ= 000100         TS.EXE= 100000         T3.PRV= 010000         U.AFLG  000064
SF.ENA= 000100         S2.PCU= 010000         TS.HLD= 002000         T3.REM= 020000         U.ATT   000022
SF.LPB= 000004         S2.RCU= 000001         TS.MSG= 020000         T3.ROV= 000040         U.BPKT= 000044
SF.MFL= 000040         S2.SRQ= 000040         TS.NRP= 010000         T3.RST= 000400         U.BUF   000024
SF.PAC= 000020         S2.WAL= 000010         TS.OUT= 000400         T3.SLV= 002000         U.CBF = 000032
SF.REA= 000010         S2.WRA= 000006         TS.RDN= 040000         T3.SWS= 000002         U.CLI   177772
SF.SER= 000001         S2.WRB= 000002         TS.RUN= 004000         UC.ALG= 000200         U.CNT   000030
SF.SVC= 000002         S3.ACR= 000001         TS.STP= 001000         UC.ATT= 000010         U.COTQ  000030
SF.UNL= 000040         S3.CTC= 000004         T$$KMG= 000000         UC.KIL= 000004         U.CTCB  000026
SLTMA   000032RG   004 S3.FDX= 000200         T$$MIN= 000000         UC.LGH= 000003         U.CTL   000004
STDDM   000034RG   004 S3.ICE= 001000         T.ACTL  000052         UC.NPR= 000100         U.CTYP  000052
SV$DUM= 000001         S3.MHE= 000400         T.ASTL  000016         UC.PWF= 000020         U.CW1   000010
SV$LOA= 000000         S3.NEC= 000020         T.ATT   000062         UC.QUE= 000040         U.CW2   000012
S$$WRG= 000000         S3.PPT= 020000         T.CPCB  000004         UD.UNS= 000000         U.CW3   000014
S$$YSZ= 007600         S3.PTH= 004000         T.DPRI  000040         UD.160= 000004         U.CW4   000016
S.ANM1= 000014         S3.RAL= 000010         T.EFLG  000022         UD.200= 000001         U.CYL = 000104
S.ANM2= 000005         S3.RES= 010000         T.IOC   000003         UD.556= 000002         U.DCB   000000
S.BFHD= 000020         S3.RUB= 040000         T.LBN   000041         UD.625= 000005         U.FCDE= 000042
S.COST  000001         S3.TAB= 000002         T.LDV   000044         UD.8K = 000006         U.FNUM= 000040
S.FATT= 000016         S3.TME= 002000         T.LNK   000000         UD.800= 000003         U.GRP = 000102
S.FDB = 000140         S3.TSY= 000040         T.MXSZ  000050         UM.CLI= 000036         U.KCSR= 000032
S.FLG   000000         S3.8BC= 000100         T.NAM   000006         UM.DSB= 000200         U.KCS6= 000034
S.FNAM= 000006         S4.ABD= 100000         T.OFF   000066         UM.NBR= 000400         U.LUIC  177774
S.FNB = 000036         S4.ANI= 000400         T.PCB   000046         UM.OVR= 000001         U.MEDI= 000040
S.FNBW= 000017         S4.AVO= 001000         T.PRI   000002         US.ABO= 000001         U.MLUN= 000050
S.FNTY= 000004         S4.BLK= 002000         T.RCVL  000012         US.BSP= 000002         U.MUP   177772
S.FTYP= 000002         S4.DEC= 004000         T.RRFL  000072         US.BSY= 000200         U.OWN   177776
S.LEN   000004         S4.DLO= 000100         T.SAST  000054         US.FOR= 000040         U.RBNS= 000112
S.NFEN= 000020         S4.EDT= 010000         T.SRCT  000071         US.FRK= 000002         U.RCTC= 000113
S.NMST  000002         S4.HFF= 000020         T.STAT  000032         US.KPF= 000001         U.RCTS= 000110
S.OWNR  000003         S4.HFL= 000007         T.ST2   000034         US.LAB= 000004         U.RED   000002
S.TSEF= 000002         S4.HHT= 000040         T.ST3   000036         US.MDE= 000002         U.RED2  000034
S1.DEC= 001000         S4.HSY= 000200         T.TCBL  000030         US.MDM= 000020         U.SCB   000020
S1.DPR= 000400         S4.RGS= 020000         T.TIO   000057         US.MNT= 000100         U.SHST= 000076
S1.DSI= 004000         S4.SFC= 040000         T.TKSZ  000060         US.OFL= 000001         U.SHUN= 000074
S1.ESC= 000002         S4.VFL= 000010         T.UCB   000026         US.PUB= 000004         U.SPC = 000036
S1.IBF= 002000         S5.ABP= 100000         T2.ABO= 000100         US.PWF= 000010         U.STS   000005
S1.IBY= 000200         S5.BCC= 020000         T2.AST= 100000         US.RED= 000002         U.ST2   000007
S1.OBY= 000100         S5.DAO= 040000         T2.CAF= 000400         US.SHR= 000001         U.SUB = 000036
S1.PTH= 000010         S5.HPC= 000014         T2.CHK= 020000         US.SPU= 000002         U.TCHP  000042
S1.RES= 010000         S5.HPO= 000020         T2.CKD= 010000         US.UMD= 000010         U.TCVP  000043
S1.RNE= 000020         S5.ITI= 000100         T2.DST= 040000         US.VV = 000001         U.TFLK  000040
S1.RNF= 020000         S5.OXF= 000040         T2.FXD= 002000         US.WCK= 000010         U.TFRQ  000036
S1.RSP= 000004         S5.RPO= 002000         T2.HLT= 000200         UU.ABO= 000400         U.TIXL  000060
S1.RST= 000001         S5.SW1= 000001         T2.REX= 001000         UU.ATN= 000100         U.TLPP  000053
S1.TNE= 040000         S5.TMM= 000002         T2.SEF= 004000         UU.AVN= 000004         U.TMTI  000047
S1.TSY= 000040         S5.VER= 010000         T2.SPN= 000004         UU.GUS= 000010         U.TRCK= 000100
S1.USI= 100000         S5.XOF= 000004         T2.STP= 000020         UU.IOS= 002000         U.TSTA  000026
S2.BEL= 020000         S5.XON= 000010         T2.WFR= 000001         UU.ONL= 000020         U.TST5  000054
```

```
                                                                            C 16
**FILE**ID**MLD11S









MM      MM   LL           DDDDDDDD          11           11         SSSSSSSS
MM      MM   LL           DDDDDDDD          11           11         SSSSSSSS
MMMM  MMMM   LL           DD      DD      1111         1111       SS
MMMM  MMMM   LL           DD      DD      1111         1111       SS
MM  MM  MM   LL           DD      DD        11           11       SS
MM  MM  MM   LL           DD      DD        11           11       SS
MM      MM   LL           DD      DD        11           11         SSSSSS
MM      MM   LL           DD      DD        11           11         SSSSSS
MM      MM   LL           DD      DD        11           11               SS
MM      MM   LL           DD      DD        11           11               SS
MM      MM   LL           DD      DD        11           11               SS       ....
MM      MM   LL           DD      DD        11           11               SS       ....
MM      MM   LLLLLLLLLL   DDDDDDDD        111111       111111     SSSSSSSS         ....
MM      MM   LLLLLLLLLL   DDDDDDDD        111111       111111     SSSSSSSS         ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                             D 16
MLD      - Microcode Loader      MACRO V05.03b Wednesday 28-Aug-85 11:31
Table of contents
```

```
                                                               D  1
UPDPSN     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE             REFERENCES

CS.CHN  = ******  GX      5-148
CS.CUG  = ******  GX      5-150
CS.DSA  = ******  GX      5-152
CS.DSC  = ******  GX      5-153
CS.DST  = ******  GX      5-151
CS.DTE  = ******  GX      5-147
CS.FLG  = ******  GX      6-193
CS.PSN  = ******  GX      5-145
CS.PVC  = ******  GX      5-149
CS.RDT  = ******  GX      5-154
CS.THI  = ******  GX      6-195
CS.TLO  = ******  GX      6-194
CS.X2P  = ******  GX      5-155
CS.X29  = ******  GX      5-157
CS.X3P  = ******  GX      5-146
C.FLG     000012         #5-133     6-193
C.LEN   = 000020         #5-136
C.NAM     000004         #5-131     6-191
C.POR     000010         #5-132     6-192
C.STS   = ******  GX      6-206
C.THI     000016         #5-135     6-195
C.TLO     000014         #5-134     6-194
DISTBL    000010 R       #5-144     6-205
F.NRBD  = ******  GX     *6-184    *6-201    *6-217
PSNDF     000000 R       #5-139     5-140     6-185
PSNLN   = 000010         #5-140     6-186
$BUFI   = ******  GX      6-217
$MVASC  = ******  GX      6-187
$OBUF   = ******  GX     *6-188     6-199
$OFDB   = ******  GX     *6-184    *6-201    *6-217
$PSIPT  = ******  GX      6-178
$PTNUM  = ******  GX      6-192     6-193     6-194     6-195
$PTOPT  = ******  GX      6-193     6-194     6-195
$PTR50  = ******  GX      6-191
$UPCHN  = ******  GX      5-148
$UPCUG  = ******  GX      5-150
$UPDSA  = ******  GX      5-152
$UPDSC  = ******  GX      5-153
$UPDST  = ******  GX      5-151
$UPDTE  = ******  GX      5-147
$UPPSN    000072 RG       5-145    #6-177
$UPPVC  = ******  GX      5-149
$UPRDT  = ******  GX      5-154
$UPX2P  = ******  GX      5-155
$UPX29  = ******  GX      5-157
$UPX3P  = ******  GX      5-146
$WDATA  = ******  GX      6-202
.BUFF   = ******  GX      6-183     6-200
.COMMA  = ******  GX     *6-190
.CONER  = ******  GX      6-222
.FLAGS  = ******  GX     *6-182
.TEMP   = ******  GX     *6-180     6-203    *6-213
```

```
                                                               E  1
UPDPSN     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55      PAGE 2
```

D 2

```
**FILE**ID**UPDRDT




UU      UU    PPPPPPPP      DDDDDDDD      RRRRRRRR      DDDDDDDD      TTTTTTTTTT
UU      UU    PPPPPPPP      DDDDDDDD      RRRRRRRR      DDDDDDDD      TTTTTTTTTT
UU      UU    PP      PP    DD      DD    RR      RR    DD      DD        TT
UU      UU    PP      PP    DD      DD    RR      RR    DD      DD        TT
UU      UU    PP      PP    DD      DD    RR      RR    DD      DD        TT
UU      UU    PP      PP    DD      DD    RR      RR    DD      DD        TT
UU      UU    PPPPPPPP      DD      DD    RRRRRRRR      DD      DD        TT
UU      UU    PPPPPPPP      DD      DD    RRRRRRRR      DD      DD        TT
UU      UU    PP            DD      DD    RR  RR        DD      DD        TT
UU      UU    PP            DD      DD    RR  RR        DD      DD        TT
UU      UU    PP            DD      DD    RR    RR      DD      DD        TT              ....
UU      UU    PP            DD      DD    RR    RR      DD      DD        TT              ....
UUUUUUUUUU    PP            DDDDDDDD      RR      RR    DDDDDDDD          TT              ....
UUUUUUUUUU    PP            DDDDDDDD      RR      RR    DDDDDDDD          TT              ....




LL                SSSSSSSS    TTTTTTTTTT
LL                SSSSSSSS    TTTTTTTTTT
LL              SS                TT
LL              SS                TT
LL              SS                TT
LL              SS                TT
LL                SSSSSS          TT
LL                SSSSSS          TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LL                      SS        TT
LLLLLLLLLL      SSSSSSSS          TT
LLLLLLLLLL      SSSSSSSS          TT
```

E 2

UPDRDT - UPDATE REMOTE DTE DEFI MACRO V05.03b Saturday 29-Jun-85 05:55

```
 55                                  ;****
 56                                  ; LOCAL MACROS
 57                                  ;****
 58
 59                                          .MACRO  PRINT   TEXT
 60                                          MOV     TEXT,R0                 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
 61                                          CALL    $PRINT                  ; PRINT MESSAGE
 62                                          .ENDM   PRINT
 63
 64                                  ;****
 65                                  ; MACRO CALLS
 66                                  ;****
 67
 68                                          .MCALL  ISTAT$,STATE$,TRAN$
 69
 70                                  ;****
 71                                  ; LOCAL DATA
 72                                  ;****
 73
 74         000074                   LA = '<                                 ; LEFT ANGLE BRACKET
 75         000076                   RA = '>                                 ; RIGHT ANGLE BRACKET
 76 000000                           TEMP:   .BLKW   1                       ; ADDRESS OF REMOTE TEMPLATE
 77 000002     074    130    130     REMDEF: .ASCIZ  /<XXXXXX>,000,000/
    000005     130    130    130
    000010     130    076    054
    000013     060    060    060
    000016     054    060    060
    000021     060    000
 78         000020                           REMLEN=<.-REMDEF>-1
 79 000023     011    122    105     REMDF:  .ASCII  <11>/REM$DF/<11>
    000026     115    044    104
    000031     106    011
 80         000010                           REMLN=.-REMDF
 81 000033                           BUFF:   .BLKB   80.                     ; OUTPUT BUFFER
 82                                          .EVEN
 83                                  ;
 84                                  ; DEFINE REMOTE TEMPLATE
 85                                  ;
 86 000000                                   .ASECT
 87         000000                           .=0
 88 000000                                   .BLKW   1                       ; C.LNK
 89 000002                                   .BLKW   1                       ; C.STS
 90 000004                           C.NAM:  .BLKW   2                       ; REMOTE NODE NAME
 91 000010                           C.ADD:  .BLKW   1                       ; REMOTE ADDRESS
 92         000012                   C.LEN=.
 93 000154                                   .PSECT
 94                                  ;
 95                                  ; DEFINE SER$DF TEMPLATE
 96                                  ;
 97 000012                                   .ASECT
 98         000000                           . = 0
 99 000000                                   .BLKW   1                       ; C.LNK
100 000002                                   .BLKW   1                       ; C.FLG
101 000004                           S$FLG:  .BLKW   1                       ; option flags
102 000006                           S$ADD:  .BLKW   1                       ; Target node address   (16 bit node address)
103 000010                           S$HST:  .BLKW   1                       ; Host node address     (16 bit node address)
104 000012                           S$HAD:  .BLKW   4                       ; Hardware address      (RAD50)
```

Table of contents

```
     6-  160     $UPROU - SETUP ROUTING DEFINITION
     7-  204     SETNUM - SET THE NEXT NUMBER
```

```
                                                .SBTTL  MACRO CALLS AND LOCAL DATA

                                                .MCALL  SAVRG,RESRG

                                        ;
                                        ; LOCAL DATA
                                        ;
                                                .NLIST  BEX
000000     011     123     105          SERDF:  .ASCII  <11>/SER$DF/<11>
        000010                                  SERLN=.-SERDF
000010     123     106     056          PH3TXT: .ASCIZ  /SF.PH3!/
                                                .EVEN
                                                .LIST   BEX

                                        ;
                                        ; DEFINE SER$DF TEMPLATE
                                        ;
000000                                          .ASECT
        000000                                  . = 0
000000                                          .BLKW   1               ; C.LNK
000002                                          .BLKW   1               ; C.FLG
000004                                  S$FLG:  .BLKW   1               ; option flags
000006                                  S$ADD:  .BLKW   1               ; Target node address    (16 bit node address)
000010                                  S$HST:  .BLKW   1               ; Host node address      (16 bit node address)
000012                                  S$HAD:  .BLKW   4               ; Hardware address       (RAD50)
000022                                  S$CIR:  .BLKW   4               ; Service circuit        (extended RAD50)
000032                                  S$DEV:  .BLKW   1               ; Service device         (extended RAD50)
000034                                  S$PSS:  .BLKW   6               ; Service password       (RAD50)
000050                                  S$DPA:  .BLKW   2               ; Dump address           (octal binary)
000054                                  S$DPC:  .BLKW   1               ; Dump count             (in k bytes)
        000056                          S$LEN=.
000020                                          .PSECT
                                        ;
                                        ; Flags definitions:
                                        ;
        000001                          SF.ADD = 1                      ; Target address specified
        000002                          SF.HST = 2                      ; Host address specified
        000004                          SF.HAD = 4                      ; Hardware address specified
        000010                          SF.CIR = 10                     ; Service circuit specified
        000020                          SF.DEV = 20                     ; Service device specified
        000040                          SF.PSS = 40                     ; Service password specified
        000100                          SF.DPA = 100                    ; Dump address specified
        000200                          SF.DPC = 200                    ; Dump count specified
        000400                          SF.PH3 = 400                    ; Service node type is Phase 3

```

```
                                                  ;****
                                                  ; LOCAL MACROS
                                                  ;****

                                                          .MACRO  PRINT   TEXT
                                                          MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                                                          CALL    $PRINT          ; PRINT MESSAGE
                                                          .ENDM   PRINT
                                                  ;
                                                  ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
                                                  ;
                                                          .MACRO  $$$PTN  ROU
                                                          .MACRO  'ROU    OFS,OCTL,OPT,?L
                                                          .IF NB <OPT>
                                                          MOV     OPT,R2
                                                          CALL    $PTOPT
                                                          BCS     L
                                                          .IFTF
                                                          MOV     OFS,R2
                                                          .IF NB <OCTL>
                                                          MOV     SP,R1
                                                          .IFF
                                                          CLR     R1
                                                          .ENDC ; IF NB <OCTL>
                                                          CALL    'ROU
                                                          .IFT  ; IF NB <OPT>
                                                  L:
                                                          .ENDC ; IF NB <OPT>
                                                          .ENDM ; 'ROU
                                                          .ENDM ; $$$PTN

 000000                                                   $$$PTN  $PTNUM
 000000                                                   $$$PTN  $PTNMB

                                                  ;****
                                                  ; MACRO  CALLS
                                                  ;****

                                                          .MCALL  ISTAT$,STATE$,TRAN$

                                                  ;****
                                                  ; LOCAL DATA
                                                  ;****

                                                          .NLIST  BEX
 000000        114        106        056  LENA:   .ASCII  /LF.ENA!/       ; LINE ENABLE SETTING
            000007                                        LENAL=.-LENA
 000007        114        106        056  LMDC:   .ASCII  /LF.MDC!/       ; LINE NEEDS MODEM CONTROL
            000007                                        LMDCL=.-LMDC
 000016        114        106        056  LMTP:   .ASCII  /LF.MTP!/       ; LINE IS MULT-POINT
            000007                                        LMTPL=.-LMTP
 000025        114        106        056  LMFL:   .ASCII  /LF.MFL!/       ; LINE IS MARKED FOR LOAD
            000007                                        LMFLL=.-LMFL
 000034        114        106        056  LTIM:   .ASCII  /LF.TIM!/       ; LINE NEEDS TIMER SUPPORT
            000007                                        LTIML=.-LTIM
 000043        114        106        056  LX2P:   .ASCII  /LF.X2P!/       ; LINE SUPPORTS X.25 LEVEL 2
            000007                                        LX2PL=.-LX2P
```

```
                                                                                 D  7
UPDSLT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE               REFERENCES

CS.SLT  = ******  GX         6-236
C.CNT     000010            #4-131       9-366
C.CT1     000012            #4-133
C.FLG     000006            #4-130       8-303       8-308       8-313       8-318       8-323       8-328       8-333       8-338
C.LEN   = 000014            #4-134
C.NAM     000004            #4-129       7-282
C.STS   = ******  GX         6-236       11-450
C.UNT     000011            #4-132       10-387
LBRO      000052 R          #4-113       4-114       8-335
LBROL   = 000007            #4-114       8-336
LENA      000000 R          #4-101       4-102       8-320
LENAL   = 000007            #4-102       8-321
LF.BRO  = ******  GX         8-333       10-427      10-433
LF.ENA  = ******  GX         8-318
LF.MDC  = ******  GX         8-303
LF.MFL  = ******  GX         8-313
LF.MTP  = ******  GX         8-308
LF.SER  = ******  GX         8-338
LF.TIM  = ******  GX         8-323
LF.X2P  = ******  GX         8-328
LMDC      000007 R          #4-103       4-104       8-305
LMDCL   = 000007            #4-104       8-306
LMFL      000025 R          #4-107       4-108       8-315
LMFLL   = 000007            #4-108       8-316
LMTP      000016 R          #4-105       4-106       8-310
LMTPL   = 000007            #4-106       8-311
LSER      000061 R          #4-115       4-116       8-340
LSERL   = 000007            #4-116       8-341
LTIM      000034 R          #4-109       4-110       8-325
LTIML   = 000007            #4-110       8-326
LX2P      000043 R          #4-111       4-112       8-330
LX2PL   = 000007            #4-112       8-331
MASTER    000070 R          #4-118       11-453
SETBRA    000732 R          #10-432
SETCNT    000540 R          #9-363
SETCTT    000614 R          #10-405
SETFLG    000242 R          #8-299
SETHTM    000642 R          #10-423
SETNAM    000204 R          #7-276
SETRPR    000670 R          #10-426
SETTYP    000774 R          #11-448
SETUNT    000566 R          #10-384
SLAVE     000074 R          #4-119
SLTKTB    000000 RG         #5-141       6-249
SLTSTB    000000 RG         #5-141       6-250
SL$BRA  = ******  GX         10-435
SL$CTM  = ******  GX         10-405
SL$HTM  = ******  GX         10-423
SL$RPR  = ******  GX         10-429
SL$TYP  = ******  GX         11-450
S$LBRA  = ******  GX         10-435
S$LCTT  = ******  GX         10-405
```

```
                                                                                 E  7
UPDSLT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 2
```

```
                                                .SBTTL  SETHTM - SET THE HELLO TIMER
                                        ;+
                                        ; *** - SETHTM - SET THE HELLO TIMER
                                        ;
                                        ; INPUTS:
                                        ;       TEMP = TEMPLATE ADDRESS
                                        ;       $OBUF = POINTER INTO OUTPUT BUFFER
                                        ;
                                        ; OUTPUTS:
                                        ;       THE HELLO TIMER IS WRITTEN TO THE OUTPUT BUFFER
                                        ;
                                        ;-

 000432                                 SETHTM:
 000432 016702  177342                          MOV     TEMP,R2                 ; GET THE TEMPLATE ADDRESS
 000436 032762  000000G 000000G                 BIT     #ST$HTM,C.STS(R2)       ; OPTION SELECTED ?
 000444 001412                                  BEQ     10$                     ; IF EQ, NOPE
 000446 016201  000012                          MOV     C.HTM(R2),R1            ; GET THE HELLO TIMER
 000452 016700  000000G                         MOV     $OBUF,R0                ; GET THE CURRENT POSITION IN THE OUTPUT BUFFER
 000456 112720  000054                          MOVB    #<',>,(R0)+             ; INSERT COMMA IN BUFFER
 000462 004767  000000G                         CALL    $CNVD                   ; CONVERT THE NUMBER TO DECIMAL ASCII
 000466 010067  000000G                         MOV     R0,$OBUF                ; UPDATE POINTER INTO OUTPUT BUFFER
 000472 000207                          10$:    RETURN

        000001                                  .END
```

```
                                                                     D  9
UPDSVC     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58        PAGE 1
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE              REFERENCES

CS.SVC  = ******  GX        6-166
C.STS   = ******  GX        6-166
F.NRBD  = ******  GX       *6-168       *6-190       *6-195
OFFLEN  = 000003           #5-139
OFFSTA    000016 R         #5-138        5-139
ONLEN   = 000002           #5-137
ONSTA     000014 R         #5-136        5-137
SVCDF     000000 R         #5-134        5-135        6-174
SVCLN   = 000014           #5-135        6-175
SV$DTE  = ******  GX        6-186
SV$PSZ  = ******  GX        6-181
SV$WSZ  = ******  GX        6-182
S$VDTE  = ******  GX        6-186
S$VFLG  = ******  GX        6-183
S$VNAM  = ******  GX        6-178        6-179
S$VOWN  = ******  GX        6-187
S$VPSZ  = ******  GX        6-181
S$VRCL  = ******  GX        6-184
S$VTIM  = ******  GX        6-185
S$VWSZ  = ******  GX        6-182
$BUFI   = ******  GX        6-195
$MVASC  = ******  GX        6-176
$OBUF   = ******  GX       *6-177       *6-180        6-188
$OFDB   = ******  GX       *6-168       *6-190       *6-195
$PTNUM  = ******  GX        6-178        6-179        6-181        6-182        6-183        6-184        6-185
$PTOPT  = ******  GX        6-181        6-182        6-186
$PTR50  = ******  GX        6-186        6-187
$SVUPD  = ******  GX        6-160       *6-162
$TMLST  = ******  GX        6-163
$UPSVC    000022 RG        #6-158
$WDATA  = ******  GX        6-191
.BUFF   = ******  GX        6-168        6-173        6-189
.COMMA  = ******  GX       *6-172
.FLAGS  = ******  GX       *6-171
.TEMP   = ******  GX       *6-169        6-192
```

```
                                                                     E  9
UPDSVC     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58        PAGE 2
```

```
                                        .SBTTL  SETCST - SET THE LINE COST
                                ;+
                                ; *** - SETCST - SET THE LINE COST
                                ;
                                ; INPUT:
                                ;       .TEMP = TEMPLATE ADDRESS
                                ;
                                ; OUTPUT:
                                ;
                                ;-

 000240                         SETCST: $PTNUM  #C.CST,,#UN$CST
 000264  000207                         RETURN
```

```
                                                 .SBTTL  SETLIN - SET LINE ID
                                        ;+
                                        ; ** SETLIN - SET LINE ID
                                        ;
                                        ; INPUT:         CURRENT TEMPLATE.
                                        ;
                                        ; OUTPUT:        VALUE LOADED INTO OUTPUT BUFFER.
                                        ;
                                        ;-

 000200                                 SETLIN: $PTR50  #C.LIN,#1                       ; DDM NAME
 000214  112777  000055  000000G                MOVB    #'-,@$OBUF                      ; SEPARATOR
 000222  005267  000000G                        INC     $OBUF                           ; INCREMENT BUFFER POINTER
 000226  005367  000000G                        DEC     .COMMA                          ; SUPPRESS COMMAS
 000232                                         $PTNUM  #C.LIN+2,OCTAL                  ; CONTROLLER
 000244  032777  000000G 000000G                BIT     #CS.UNI,@.FLAGS                 ; UNIT NUMBER IN LINE ID ?
 000252  001414                                 BEQ     10$                             ; IF EQ, NONE
 000254  112777  000055  000000G                MOVB    #'-,@$OBUF                      ; ELSE BUFFER SEPARATOR,
 000262  005267  000000G                        INC     $OBUF                           ; BUMP BUFFER POINTER
 000266  005367  000000G                        DEC     .COMMA                          ; SUPPRESS COMMAS
 000272                                         $PTNUM  #C.LIN+4,OCTAL                  ; OUTPUT UNIT NUMBER
 000304  000207                         10$:    RETURN


         000001                                 .END
```

```
UPDX29     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59      PAGE 2
MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME       REFERENCES

EPRINT           #4-78
ERROR$           #4-67
PRINT            #4-62
$PTNMB           #4-111
$PTNUM           #4-110        6-152        6-153
$PTR50           #4-115
$$$PTN           #4-90         4-110        4-111
```

```
**FILE**ID**UPDX3P
```

```
                        .ENABL  LC
                        .NLIST  BEX
                        .TITLE  MLD     - Microcode Loader
                        .IDENT  /V05.00/

;+
; Copyright (C) 1982, 1983, 1984, 1985 by
; Digital Equipment Corporation, Maynard, Mass.
;
; This software is furnished under a license for use only on a
; single computer system and  may  be  copied  only  with  the
; inclusion of the above copyright notice.  This software,  or
; any other copies thereof, may not be provided  or  otherwise
; made available to any other person except for  use  on  such
; system and to one who agrees to these license  terms.  Title
; to  and  ownership of the software shall at all times remain
; in DEC.
;
; The information in this document is subject to change without
; notice and should not be construed as a commitment by Digital
; Equipment Corporation.
;
; DEC assumes no responsibility for the use or reliability  of
; its software on equipment which is not supplied by DEC.
;
; Module Description
;
;       DECnet/RSX Microcode Loader
;
; Distributed Systems Software Engineering
;
; Ident History:
;
; 1.00  5-Oct-82        R. Levesque
;       Original development
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
                                   .SBTTL  PWRAST  - Powerfail Recovery AST
                                   .SBTTL  ABOAST  - Abort Cleanup AST
                           ;+
                           ;                               *** PWRAST/ABOAST ***
                           ;
                           ; Function        To perform any termination and cleanup of in-progress or queued
                           ;                 load requests at the time of a system powerfail or a task abort
                           ;                 from a privileged terminal.
                           ;
                           ; Call Format:    Not applicable
                           ;
                           ; Inputs:         No explicit inputs
                           ;
                           ; Outputs:        Current load sequence ended,  load request listhead in AUX is
                           ;                 emptied with a "halt load" dispatch for each CCB dequeued.
                           ;
                           ; Registers:      R0 through R5 destroyed
                           ;
                           ; Notes:          When RSX systems recover from powerfail,  any active tasks
                           ;                 with powerfail ASTs are dispatched to before any device
                           ;                 drivers are dispatched to at their powerfail recovery
                           ;                 entry points.
                           ;
                           ;                 MLD stays in system state until all load requests have been
                           ;                 terminated.  This is so that AUX will not queue more CCBs
                           ;                 while MLD is trying to clean up and exit.
                           ;-

001152                     PWRAST:
001152                     ABOAST: .IFT                                    ;; If system is RSX-11M ...

001152                             CLOSE$  #FDB                            ;; Close the microcode file

                                   .IFTF                                   ;; For any operating system ...

001162                             CALL    $SWSTK,40$                      ;; Switch to system state

                                   .IF     DF      N$$UMR                  ;; If system has UMRs ...

001166  016704  001364'            MOV     LDBPTR,R4                       ;; Get address of CCB/LDB
001172  001402                     BEQ     5$                              ;; Branch if none
001174                             CALL    @LDBRT                          ;; Return the CCB and LDB to pool

001200                     5$:     .ENDC                                   ;; End UMR conditional

001200  016704  001376'            MOV     CURQST,R4                       ;; Get address of current CCB
001204  001010                     BNE     20$                             ;; Branch if none

001206  017700  176602     10$:    MOV     @TKTCB,R0                       ;; Get our TCB address
001212  062700  000012             ADD     #T.RCVL,R0                      ;; Advance to receive listhead
001216                             CALL    @QRMVF                          ;; Dequeue a load request
001222  103421                     BCS     30$                             ;; Branch if none left
001224  010104                     MOV     R1,R4                           ;; Copy CCB address

001226                     20$:    CALL    @STDDM                          ;; Get PDV index and line table address
001232  010403                     MOV     R4,R3                           ;; Copy CCB address
001234  062703  000012             ADD     #C.STS,R3                       ;; Advance to status field
```

```
                                                             D 15
MLD     - Microcode Loader      MACRO V05.03b Wednesday 28-Aug-85 11:26  Page 18-5
Symbol table

U.TST6  000056           U2.HFF= 010000           X$$DBT= 000000           Z.DAT   000016           $QRMVF= ****** GX
U.TTAB  000050           U2.HLD= 000040           ZF.COU= 001000           Z.DSP   000000           $RDBSZ= ****** GX
U.TTYP  000046           U2.LOG= 000400           ZF.DDM= 000001           Z.FLG   000010           $SLTMA= ****** GX
U.TUX   000024           U2.LWC= 000001           ZF.DIA= 004000           Z.LEN = 000016           $STDDM= ****** GX
U.UHVR= 000107           U2.L3S= 000004           ZF.DLC= 000002           Z.LLN   000006           $TKTCB= ****** GX
U.UIC   000044           U2.L8S= 010000           ZF.DVP= 100000           Z.MAP   000020           $$$   = 000062
U.UNFL= 000052           U2.NEC= 004000           ZF.INI= 040000           Z.NAM   000004           $$$ARG= 000002
U.UNIT  000006           U2.PRV= 000010           ZF.KMX= 000020           Z.PCB   000012           $$$OST= 000004
U.UNSZ= 000114           U2.RMT= 020000           ZF.LLC= 000004           Z.SCH   000007           $$$T1 = 000067
U.UNTI= 000060           U2.R04= 100000           ZF.LMC= 000100           $BLXIO= ****** GX        .CLOSE= ****** G
U.USVR= 000106           U2.SCS= 000004           ZF.MAN= 020000           $CBTA = ****** GX        .FINIT= ****** G
U.UTIL= 000036           U2.SLV= 000200           ZF.MFL= 000010           $CCBRT= ****** GX        .FSRCB= ****** G
U.VCB = 000034           U2.VT5= 000002           ZF.MTM= 000400           $DIV  = ****** GX        .OPEN = ****** G
U.VSER= 000120           U2.7CH= 010000           ZF.MUX= 000040           $EDMSG= ****** GX        .PPASC= ****** GX
U.2MED= 000070           U3.OPA= 100000           ZF.PSE= 002000           $LDBGT= ****** GX        .READ = ****** G
U2.AT.= 000020           U3.PAR= 040000           ZF.SLI= 010000           $LDBRT= ****** GX        .WAIT = ****** G
U2.CRT= 002000           U3.UPC= 020000           ZF.TIM= 000200           $NTUIC= ****** GX        .$$$$.= 000034
U2.DH1= 100000           U4.CR = 000100           ZF.X3P= 000000           $PARHD= ****** GX        ...PC1= 000144R     002
U2.DJ1= 040000           V$$CTR= 001000           ZS.ASN= 100000           $PDSPL= ****** GX        ...PC2= 000304R     002
U2.DZ1= 000100           WAIT    001360R      002 ZS.BSY= 140000           $PDVTA= ****** GX        ...TPC= 000020
U2.ESC= 001000           WORKSP  001406R      002 Z.AVL   000014

. ABS.  177776    000    (RW,I,GBL,ABS,OVR)
        000000    001    (RW,I,LCL,REL,CON)
MLDDAT  002006    002    (RW,D,LCL,REL,OVR)
$$FSR1  000000    003    (RW,D,GBL,REL,OVR)
MLDCOD  001634    004    (RO,I,LCL,REL,OVR)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 147
Work  file writes: 114
Size of work file: 24854 Words  ( 98 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:02:03.58
DB2:MLD.T34,[132,134]MLD/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]T34,MLD
```

```
                                                  E 15
MLD       CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 1
```

Table of contents

```
     6-   49    Macro Definitions
     7-   93    Equated Symbols and Literals
     8-  161    Load Control Data
     9-  249    MLD      - Microcode Loader Main Routine
    10-  471    DSPDRV   - Dispatch to Device Driver
    11-  579    LOGEVT   - Log a Network Event
    12-  620    PWRAST   - Powerfail Recovery AST
    12-  621    ABOAST   - Abort Cleanup AST
    16-  813    LOCPAR   - Locate Microcode Partition
    17-  891    SRNAM    - Search For Named Partition
    18-  931    SLOAD    - Load Control for RSX-11S Systems
```

E 1

```
UPDPSN     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 2
MACPO CROSS REFERENCE                                    CREF    04.00
MACRO NAME        REFERENCES

EPRINT            #4-78
ERROR$            #4-67
PRINT             #4-62
$PTNMB            #4-111
$PTNUM            #4-110      6-192      6-193      6-194      6-195
$PTR50            #4-115      6-191
$$$PTN            #4-90       4-110      4-111
```

F 1

```
**FILE**ID**UPDPVC
```

```
Table of contents

     4-   57     MACRO CALLS AND LOCAL DATA
     4-   58     MACRO DEFINITIONS
     5-  130     LOCAL DATA
     6-  150     $UPRDT - UPDATE DESTINATION DESCRIPTOR
```

```
000022                          S$CIR:  .BLKW   4               ; Service circuit       (extended RAD50)
000032                          S$DEV:  .BLKW   1               ; Service device        (extended RAD50)
000034                          S$PSS:  .BLKW   6               ; Service password      (RAD50)
000050                          S$DPA:  .BLKW   2               ; Dump address          (octal binary)
000054                          S$DPC:  .BLKW   1               ; Dump count            (in k bytes)
        000056                  S$LEN=.
000154                                  .PSECT


```

```
                                .TITLE  UPDROU - CFE UPDATE ROUTING DEFINITION IN CETAB
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       STATE TABLE TO PARSE THE ROUTING DEFINITIONS
                        ;
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;

```

```
     93                                                 .SBTTL  $UPSER - SETUP REMOTE DEFINITION
     94                                         ;+
     95                                         ; *** - $UPSER - SETUP REMOTE DEFINITION
     96                                         ;
     97                                         ; INPUT:
     98                                         ;       R5      - TEMPLATE ADDRESS
     99                                         ;
    100                                         ; OUTPUT:
    101                                         ;       C-BIT = SUCCESS/FAILURE
    102                                         ;
    103                                         ;-
    104
    105 000020                                  $UPSER::
    106 000020  004567  000000G                         JSR     R5,$SAVRG       ; SAVE R3-R5
    107 000024  016746  000000C                         MOV     $OFDB+F.NRBD+2,-(SP); SAVE OUTPUT BUFFER
    108 000030  012767  000000G 000000C                 MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
    109 000036  010567  000000G                         MOV     R5,.TEMP        ; SAVE TEMPLATE ADDRESS
    110 000042  062705  000004                          ADD     #S$FLG,R5       ; POINT TO FLAGS
    111 000046  010567  000000G                         MOV     R5,.FLAGS       ; SAVE ADDRESS OF FLAGS WORD
    112 000052  012700  000000G                         MOV     #.BUFF,R0       ; POINT AT START OF OUTPUT BUFFER
    113 000056  012701  000000'                         MOV     #SERDF,R1       ; POINT AT START OF SER DEFINITION
    114 000062  012702  000010                          MOV     #SERLN,R2       ; GET LENGTH OF SER DEFINITION
    115 000066  004767  000000G                         CALL    $MVASC          ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
    116 000072  010067  000000G                         MOV     R0,$OBUF        ; SAVE CURRENT POSITION IN OUTPUT BUFFER
    117
    118 000076  004567  000164                          JSR     R5,PUTNOD       ; OUTPUT REMOTE NODE ADDRESS
    119 000102  000006  000001                          .WORD   S$ADD,SF.ADD    ;  ...
    120
    121 000106  004567  000154                          JSR     R5,PUTNOD       ; OUTPUT HOST NODE ADDRESS
    122 000112  000010  000002                          .WORD   S$HST,SF.HST    ;  ...
    123
    124 000116  004567  000244                          JSR     R5,PUTR50       ; OUTPUT SERVICE CIRCUIT
    125 000122  000022  000004  000010                  .WORD   S$CIR,4,SF.CIR  ;  ...
    126
    127 000130  004567  000232                          JSR     R5,PUTR50       ; SERVICE DEVICE
    128 000134  000032  000001  000020                  .WORD   S$DEV,1,SF.DEV  ;  ...
    129
    130 000142  004567  000220                          JSR     R5,PUTR50       ; PASSWORD
    131 000146  000034  000006  000040                  .WORD   S$PSS,6,SF.PSS ;  ...
    132
    133 000154  004567  000262                          JSR     R5,PUTOCT       ; DUMP ADDRESS
    134 000160  000050  060210  000004                  .WORD   S$DPA,8.+<40*4>+<4000*14>,4,SF.DPA
        000166  000100
    135
    136 000170  004567  000246                          JSR     R5,PUTOCT       ; DUMP COUNT
    137 000174  000054  060110  000002                  .WORD   S$DPC,8.+<40*2>+<4000*14>,2,SF.DPC
        000202  000200
    138
    139 000204  004567  000156                          JSR     R5,PUTR50       ; HARDWARE ADDRESS
    140 000210  000012  000004  000004                  .WORD   S$HAD,4,SF.HAD  ;  ...
    141
    142 000216  004567  000266                          JSR     R5,PUTFLG       ; OPTIONAL FLAGS
    143 000222  000010' 000400                          .WORD   PH3TXT,SF.PH3   ; PHASE 3 TARGET NODE VERSION
    144 000226  000000                                  .WORD   0               ; THAT'S ALL, FOLKS
    145
    146 000230  016701  000000G                         MOV     $OBUF,R1        ; CALCULATE LENGTH OF BUFFER
    147 000234  124127  000054                  10$:    CMPB    -(R1),#<',>     ; TRAILING COMMA ?
    113 000052     114     106     056  LBRO:   .ASCII  /LF.BRO!/         ; LINE IS A BROADCAST CHANNEL
    114         000007                          LBROL=.-LBRO
    115 000061     114     106     056  LSER:   .ASCII  /LF.SER!/         ; LINE HAS SERVICE DISABLED
    116         000007                          LSERL=.-LSER
    117                                         .EVEN
    118 000070  050573  076732          MASTER: .RAD50  /MASTER/
    119 000074  074241  105110          SLAVE:  .RAD50  /SLAVE/
    120                                         .EVEN
    121                                         .LIST   BEX
    122                                 ;
    123                                 ; DEFINE THE SLT TEMPLATE
    124                                 ;
    125 000000                                  .ASECT
    126         000000                          .=0
    127 000000                                  .BLKW   1                 ; C.LNK
    128 000002                                  .BLKW   1                 ; C.STS
    129 000004                          C.NAM:  .BLKW   1                 ; THE DDM PROCESS ASSOCIATED WITH THIS  LINE
    130 000006                          C.FLG:  .BLKW   1                 ; FLAGS WORD
    131 000010                          C.CNT:  .BLKB   1                 ; CONTROLLER NUMBER OF THE ASSOCIATED DDM
    132 000011                          C.UNT:  .BLKB   1                 ; THE UNIT NUMBER OF THE ASSOCIATED DDM
    133 000012                          C.CTT:  .BLKW   1                 ; THE COUNTER TIMER (X.25 LINES ONLY)
    134         000014                  C.LEN=.
    135 000100                                  .PSECT
```

```
                                                                  E  7
UPDSLT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE             REFERENCES

S$LFLG  = ******  GX        10-427     10-433
S$LHTM  = ******  GX        10-423
S$LRPR  = ******  GX        10-429
S$LTYP  = ******  GX        11-454
$ALPHA  = 000022           #5-141
$ANY    = 000020           #5-141
$BLANK  = 000006           #5-141
$CATS   = ******  GX         7-280
$DIGIT  = 000024           #5-141
$DNUMB  = 000014           #5-141
$EOS    = 000012           #5-141
$ERROR  = ******  GX       *6-253
$EXIT   = 000000           #5-141
$FAIL   = 177777           #5-141
$GPRM   = ******             5-141
$LAMDA  = 000000           #5-141
$MOVPT  = ******  GX         7-277      9-364     10-385
$MOVST  = ******  GX         6-246
$MVASC  = ******  GX         8-307      8-312      8-317      8-322      8-327      8-332      8-337      8-342
$NUMBR  = 000002           #5-141
$OBUF   = ******  GX         8-302      8-343     *8-347
$PTNUM  = ******  GX        10-405     10-423     10-429     10-435
$PTOPT  = ******  GX        10-405     10-423     10-429     10-435
$PTR50  = ******  GX        11-459
$RAD50  = 000016           #5-141
$RONLY  = ******             5-141      5-141      5-141
$STRNG  = 000004           #5-141
$SUBXP  = 000010           #5-141
$TMLST  = ******  GX         6-232
$UPSLT    000100 RG        #6-231
$$$FLG  = 177777           #5-141
$$$KEY  = 177777           #5-141
.COMMA  = ******  GX       *6-245     *11-463
.CONER  = ******  GX         6-259
.FLAGS  = ******  GX       *6-244
.PNUMB  = ******  GX         9-366     10-387
.PSTPT  = ******  GX         7-278
.TEMP   = ******  GX       *6-242      7-281      8-301      9-365     10-386     10-426     10-432     11-449    *11-457
                          *11-460

.TPARS  = ******  GX         6-251
```

```
                                                                  F  7
UPDSLT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 3
```

```
                                                                         E  8
UPDSTA - CFE UPDATE STATION DEF MACRO V05.03b Saturday 29-Jun-85 05:57  Page 11-1
Symbol table

APRNUM  000064R     002 SENA    000002R         SMFL    000011R         $BUFI = ****** GX       $NUMBR= 000002
BITS    000124R     002 SENAL = 000007          SMFLL = 000007          $CLNUP= ****** GX       $OBUF = ****** GX
CS.MOD= ****** GX       SETADD  000150R         SSER    000027R         $CNVD = ****** GX       $OFDB = ****** GX
CS.STA= ****** GX       SETAPR  000226R         SSERL = 000007          $CNVO = ****** GX       $RAD50= 000016
C.ADD   000005          SETCST  000200R         SSVC    000020R         $DIGIT= 000024          $STRNG= 000004
C.APR   000010          SETFLG  000254R         SSVCL = 000007          $DNUMB= 000014          $SUBXP= 000010
C.CST   000007          SETHTM  000432R         STACST  000046R     002 $EOS  = 000012          $TMLST= ****** GX
C.FLG   000006          SFLAG   000026R     002 STAKTB  000000RG    003 $ERROR= ****** GX       $UPSTA  000036RG
C.HTM   000012          SFS     000104R     002 STASTB  000000RG    002 $EXIT = 000000          $$$FLG= 177777
C.LEN = 000016          SF.ENA= ****** GX       ST$HTM= ****** GX       $FAIL = 177777          $$$KEY= 000000
C.STA   000004          SF.MFL= ****** GX       TEMP    000000R         $LAMDA= 000000          $$$STA= 000112R     002
C.STS = ****** GX       SF.SER= ****** GX       $ALPHA= 000022          $MOVPT= ****** GX       $$$TMP= 000000R     004
EXIT    000076R     002 SF.SVC= ****** GX       $ANY  = 000020          $MOVST= ****** GX       .CONER= ****** GX
F.NRBD= ****** GX       SKIP    000112R     002 $BLANK= 000006          $MVASC= ****** GX       .TPARS= ****** GX


. ABS.  000016    000   (RW,I,GBL,ABS,OVR)
        000474    001   (RW,I,LCL,REL,CON)
$STATE  000130    002   (RW,D,LCL,REL,CON)
$KTAB   000002    003   (RW,D,LCL,REL,CON)
$KSTR   000003    004   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10424 Words  ( 41 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:27.53
SY:UPDSTA.V2,[132,134]UPDSTA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDSTA
```

```
                                                         F  8
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 1
```

```
UPDSVC     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME      REFERENCES

EPRINT          #4-79
ERROR$          #4-68
PRINT           #4-63
$PTNMB          #4-112
$PTNUM          #4-111     6-178      6-179      6-181      6-182      6-183      6-184      6-185
$PTR50          #4-116     6-186      6-187
$$$PTN          #4-91      4-111      4-112
```

```
**FILE**ID**UPDUNT
```

```
                                      .SBTTL  SETDPR - SET THE LINE DEAD POLLING RATIO
                              ;+
                              ; *** - SETDPR - SET THE LINE DEAD POLLING RATIO
                              ;
                              ; INPUT:
                              ;         .TEMP = TEMPLATE ADDRESS
                              ;
                              ; OUTPUT:
                              ;
                              ;-

000266                        SETDPR: $PTNUM  #C.DPR,,#UN$DPR
000312  000207                        RETURN
```

```
UPDX2P - UPDATE X.25 LEVEL 2 CH MACRO V05.03b Saturday 29-Jun-85 05:58  Page 6-1
Symbol table

CS.UNI= ****** GX         C.RTC    000016         X2PDF    000000R        $OFDB  = ****** GX       .BUFF  = ****** GX
C.BSZ    000014           C.RTT    000020         X2PLN  = 000010         $PTNUM= ****** GX        .COMMA= ****** GX
C.HBT    000022           C.WSZ    000012         $BUFI  = ****** GX      $PTR50= ****** GX        .FLAGS= ****** GX
C.LEN  = 000024           F.NRBD= ****** GX       $MVASC= ****** GX       $UPX2P   000010RG        .TEMP  = ****** GX
C.LIN    000004           SETLIN   000200R        $OBUF  = ****** GX      $WDATA= ****** GX

. ABS.   000024     000    (RW,I,GBL,ABS,OVR)
         000306     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8783 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.17
SY:UPDX2P.V2,[132,134]UPDX2P/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDX2P
```

```
**FILE**ID**UPDX3P
```

```
UU      UU   PPPPPPPP     DDDDDDDD     XX      XX       333333     PPPPPPPP
UU      UU   PPPPPPPP     DDDDDDDD     XX      XX       333333     PPPPPPPP
UU      UU   PP      PP   DD      DD   XX      XX   33      33   PP      PP
UU      UU   PP      PP   DD      DD   XX      XX   33      33   PP      PP
UU      UU   PP      PP   DD      DD     XX  XX           33   PP      PP
UU      UU   PP      PP   DD      DD     XX  XX           33   PP      PP
UU      UU   PPPPPPPP     DD      DD       XX           33     PPPPPPPP
UU      UU   PPPPPPPP     DD      DD       XX           33     PPPPPPPP
UU      UU   PP           DD      DD     XX  XX           33   PP
UU      UU   PP           DD      DD     XX  XX           33   PP
UU      UU   PP           DD      DD   XX      XX   33      33   PP              ....
UU      UU   PP           DD      DD   XX      XX   33      33   PP              ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX     333333     PP              ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX     333333     PP              ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

UPDX3P - UPDATE X.25 LEVEL 3 CH MACRO V05.03b Saturday 29-Jun-85 05:59
Table of contents

```
     49                                             .SBTTL  Macro Definitions
     50
     51
     52                                     ;       System Macros Used
     53
     54                                             .IF     DF      R$$MPL
     55
     56                                             .MCALL  KRBDF$, RMAF$S, STAF$S, DIR$, GIN$
     57
     58                                             KRBDF$
     59
     60                                             .ENDC
     61
     62                                             .IF     NDF     R$$11S
     63
     64                                             .MCALL  FSRSZ$, OPEN$R, CLOSE$, READ$,  WAIT$,  FDBDF$
     65                                             .MCALL  FDAT$A, FDRC$A, FDBK$A, FDOP$A, FINIT$, UCBDF$, DCBDF$
     66
     67 000000                                      UCBDF$
     68 000000                                      DCBDF$
     69
     70                                             .IFF
     71
     72                                             .MCALL  PCBDF$
     73
     74                                             PCBDF$
     75
     76                                             .IFTF
     77
     78                                             .MCALL  EXIT$S, SPRA$S, SREA$S, QIOW$S, ALUN$S
     79                                             .MCALL  TCBDF$, CCBDF$, EVLDF$, PDVDF$, SLTDF$
     80
     81 000000                                      TCBDF$
     82 000000                                      CCBDF$
     83 000000                                      EVLDF$
     84 000000                                      PDVDF$
     85 000000                                      SLTDF$
     86
     87                                             .IF     DF      N$$UMR
     88
     89                                             .MCALL  MRKT$,  STSE$,  DIR$
     90
     91                                             .ENDC
```

```
    677 001240  012713  101000                  MOV     #CS.ERR+CS.FTL,(R3)     ;; Tell driver a fatal error occurred
    678 001244  012743  000026                  MOV     #FC.MLD+FS.HLT,-(R3)    ;; (Sub)function = load halted
    679 001250  010446                          MOV     R4,-(SP)                ;; Save CCB address
    680 001252                                  CALL    @PDSPL                  ;; Dispatch to driver w/priority
    681 001256  012604                          MOV     (SP)+,R4                ;; Recover CCB address
    682 001260                                  CALL    @CCBRT                  ;; Return CCB to pool
    683 001264  000750                          BR      10$                     ;; Dequeue another load request
    684
    685 001266                          30$:    RETURN                          ;; Return to user state at "40$"
    686 001270  005067  001376'         40$:    CLR     CURQST                  ;  No more requests outstanding
    687 001274                                  EXIT$S                          ;  Program exit
    688
    689                                         .IFT                            ; If system is RSX-11M  ...
```

```
                                                                   E 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 1
SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE             REFERENCES

ABOAST     001152 R         9-319       #12-649
AC$DNT  =  000002          #6-83
AC$X25  =  000001          #6-83
AE$CIR  =  000003          #6-83
AE$LIN  =  000001          #6-83
AE$MOD  =  000004          #6-83
BLOCK      001372 R        #8-210       *14-767     *14-768     *15-793      15-794
BLXIO      000004 RG       #8-230        10-529
CCBRT      000016 RG       #8-238        9-355       12-682
CF.LB   =  100000           15-804
CL$ASZ  =  010500          #6-83         7-143
CL$DLL  =  000500          #6-83         6-83        6-83        6-83        6-83        6-83        6-83        6-83        6-83
                            6-83         6-83
CL$ECL  =  000300          #6-83         6-83
CL$LDN  =  010400          #6-83         6-83
CL$MAN  =  000000          #6-83         6-83        6-83        6-83
CL$PAZ  =  034100          #6-83         6-83        6-83
CL$PLH  =  034000          #6-83         6-83        6-83        6-83        6-83
CL$PLL  =  000600          #6-83
CL$PRT  =  034200          #6-83
CL$ROU  =  010000          #6-83         6-83        6-83
CL$SES  =  000200          #6-83         6-83        6-83
CL$SGE  =  035000          #6-83         6-83        6-83
CL$SSE  =  035100          #6-83         6-83        6-83        6-83        6-83
CL$TRN  =  000400          #6-83         6-83        6-83        6-83        6-83        6-83        6-83        6-83        6-83
                            6-83         6-83        6-83        6-83        6-83        6-83        6-83        6-83        6-83
                            6-83
CL$XL2  =  013700          #6-83
CL$XL3  =  013600          #6-83         6-83
CL$X2S  =  013500          #6-83         6-83
CS.ERR  =  100000           9-464        10-571      12-677
CS.FTL  =  001000           9-444        12-677      13-725
CTLNUM     001402 R        #8-213       *9-405       11-600
CURQST     001376 R        #8-211       *9-310       9-353      *9-356      *9-363       9-388       9-442       9-462       12-665
                           *12-686
CVTPRM  =  014010          #7-109        9-295
C.BUF1     000014           9-333        9-334       9-429      *10-513      10-518      10-519     *10-532     *10-558      15-800
C.CNT1     000020           10-521      *10-531     *10-559
C.FNC      000010          *9-335        9-364      *9-447      *9-463       10-546
C.LIN      000006           9-390       *9-395
C.MOD      000011           10-508       10-550
C.STS      000012          *9-465        10-556      12-676
DDMTBL     001404 R        #8-214       *9-404       10-544
DEVERR     000051 R        #7-148        10-573
DL$AST  =  000002          #6-83
DL$HLT  =  000000          #6-83
DL$IST  =  000001          #6-83
DL$MAI  =  000004          #6-83
DL$OFF  =  000001          #6-83
DL$ON   =  000000          #6-83
DL$RUN  =  000003          #6-83
DL$SHU  =  000002          #6-83
```

```
                                                                   F 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 2
```

```
        .ENABL  LC
        .NLIST  BEX
        .TITLE  MLD     - Microcode Loader
        .IDENT  /V05.00/

;+
; Copyright (C) 1982, 1983, 1984, 1985 by
; Digital Equipment Corporation, Maynard, Mass.
;
; This software is furnished under a license for use only on a
; single computer system and may  be  copied  only  with  the
; inclusion of the above copyright notice.  This software,  or
; any other copies thereof, may not be provided  or  otherwise
; made available to any other person except for  use  on  such
; system and to one who agrees to these license  terms.  Title
; to  and  ownership of the software shall at all times remain
; in DEC.
;
; The information in this document is subject to change without
; notice and should not be construed as a commitment by Digital
; Equipment Corporation.
;
; DEC assumes no responsibility for the use or reliability  of
; its software on equipment which is not supplied by DEC.
;
; Module Description
;
;       DECnet/RSX Microcode Loader
;
; Distributed Systems Software Engineering
;
; Ident History:
;
; 1.00  5-Oct-82        R. Levesque
;       Original development
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
**FILE**ID**UPDPVC




UU      UU   PPPPPPPP     DDDDDDDD     PPPPPPPP     VV      VV     CCCCCCCC
UU      UU   PPPPPPPP     DDDDDDDD     PPPPPPPP     VV      VV     CCCCCCCC
UU      UU   PP      PP   DD      DD   PP      PP   VV      VV   CC
UU      UU   PP      PP   DD      DD   PP      PP   VV      VV   CC
UU      UU   PP      PP   DD      DD   PP      PP   VV      VV   CC
UU      UU   PP      PP   DD      DD   PP      PP   VV      VV   CC
UU      UU   PPPPPPPP     DD      DD   PPPPPPPP     VV      VV   CC
UU      UU   PPPPPPPP     DD      DD   PPPPPPPP     VV      VV   CC
UU      UU   PP           DD      DD   PP           VV      VV   CC
UU      UU   PP           DD      DD   PP           VV      VV   CC
UU      UU   PP           DD      DD   PP             VV  VV     CC                ....
UU      UU   PP           DD      DD   PP             VV  VV     CC                ....
UUUUUUUUUU   PP           DDDDDDDD     PP               VV         CCCCCCCC        ....
UUUUUUUUUU   PP           DDDDDDDD     PP               VV         CCCCCCCC        ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
UPDPVC - UPDATE PERMANENT VIRTU MACRO V05.03b Saturday 29-Jun-85 05:55
Table of contents
```

```
                                        .TITLE  UPDRDT - UPDATE REMOTE DTE DEFINITION IN CETAB
                                        .IDENT  /V05.00/
                                        .ENABL  LC

                                ;
                                ; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ;       UPDATE REMOTE DTE DEFINITION IN CETAB
                                ;
                                ;
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  14-AUG-81
                                ;       DECNET-11M/S V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 3.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;
```

```
    115                             ;****
    116                             ; INITIALIZE TABLE GENERATION
    117                             ;****
    118
    119
    120 000154                              ISTAT$  REMSTB,REMKTB
    121
    122
    123 000154                              STATE$
    124 000154                               TRAN$  LA,,$MOVPT
    125
    126 000154                              STATE$
    127 000154                               TRAN$  $STRNG,,SETNAM
    128
    129 000154                              STATE$
    130 000154                               TRAN$  RA,,$MOVPT
    131
    132 000154                              STATE$
    133 000154                               TRAN$  <',>,,$MOVPT
    134
    135 000154                              STATE$
    136 000154                               TRAN$  $NUMBR,ADD2,SETARE
    137
    138 000154                              STATE$  ADD2
    139 000154                               TRAN$  <',>,,$MOVPT
    140
    141 000154                              STATE$
    142 000154                               TRAN$  $NUMBR,,SETADD
    143                             ;
    144                             ; EXIT STATE
    145                             ;
    146 000154                              STATE$  EXIT
    147 000154                               TRAN$  $EOS,$EXIT
    148
    149 000154                              STATE$
```

```
                        ;****
                        ; LOCAL MACROS
                        ;****

                                .MACRO  PRINT   TEXT
                                MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                                CALL    $PRINT          ; PRINT MESSAGE
                                .ENDM   PRINT

                        ;****
                        ; MACRO CALLS
                        ;****

                                .MCALL  ISTAT$,STATE$,TRAN$

                        ;****
                        ; LOCAL DATA
                        ;****

000000                  TEMP:   .BLKW   1               ; ADDRESS OF REMOTE TEMPLATE

                        ;
                        ; DEFINE ROUTING TEMPLATE
                        ;
000000                          .ASECT
        000000                  .=0
000000                          .BLKW   1               ; C.LNK
000002                          .BLKW   1               ; C.STS
000004                  C.ADD:  .BLKW   1               ; MAXIMUM NODE ADDRESS
000006                  C.CST:  .BLKW   1               ; MAXIMUM COST ALLOWED
000010                  C.HOP:  .BLKW   1               ; MAXIMUM HOPS ALLOWED
000012                  C.RTM:  .BLKW   1               ; ROUTING TIMER
000014                  C.NA:   .BLKW   1               ; NUMBER OF AREAS
000016                  C.NBEA: .BLKW   1               ; NUMBER BRO ENDNODE ADJACENCIES
000020                  C.AHOP: .BLKW   1               ; AREA HOPS
000022                  C.ACST: .BLKW   1               ; AREA COST
000024                  C.L2TM: .BLKW   1               ; LEVEL 2 TIMER
        000026          C.LEN=.
000002                          .PSECT
```

```
    148 000240  001775                          BEQ     10$               ; IF EQ, YES
    149 000242  005201                          INC     R1                ; ELSE BACKED UP TOO FAR
    150
    151 000244  162701  000000G                 SUB     #.BUFF,R1         ; ...
    152 000250  010167  000000C                 MOV     R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
    153 000254  004767  000000G                 CALL    $WDATA            ; WRITE THE RECORD TO THE OUTPUT FILE
    154 000260  012667  000000C                 MOV     (SP)+,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE ''END$DF'' LINE
    155 000264  000207                          RETURN
```

```
   137                                 ;****
   138                                 ; INITIALIZE TABLE GENERATION
   139                                 ;****
   140
   141 000100                                  ISTAT$  SLTSTB,SLTKTB
   142
   143 000100                                  STATE$
   144 000100                                   TRAN$  $RAD50,,SETNAM
   145
   146 000100                                  STATE$
   147 000100                                   TRAN$  <',>,,$MOVPT
   148
   149 000100                                  STATE$
   150 000100                                   TRAN$  $RAD50,,$MOVPT
   151
   152 000100                                  STATE$
   153 000100                                   TRAN$  <',>,,$MOVPT
   154
   155 000100                                  STATE$
   156 000100                                   TRAN$  $RAD50,,$MOVPT
   157
   158 000100                                  STATE$
   159 000100                                   TRAN$  <',>,,$MOVPT
   160
   161 000100                                  STATE$
   162 000100                                   TRAN$  !LFS
   163
   164 000100                                  STATE$
   165 000100                                   TRAN$  !BITS,,SETFLG
   166
   167 000100                                  STATE$
   168 000100                                   TRAN$  $NUMBR,,SETCNT
   169
   170 000100                                  STATE$
   171 000100                                   TRAN$  <',>,,$MOVPT
   172
   173 000100                                  STATE$
   174 000100                                   TRAN$  $NUMBR,,SETUNT
   175
   176 000100                                  STATE$
   177 000100                                   TRAN$  $LAMDA,,SETTYP
   178 000100                                  STATE$
   179 000100                                   TRAN$  $LAMDA,,SETCTT
   180 000100                                  STATE$
   181 000100                                   TRAN$  $LAMDA,,SETHTM
   182 000100                                  STATE$
   183 000100                                   TRAN$  $LAMDA,,SETBRA
   184 000100                                  STATE$
   185 000100                                   TRAN$  $LAMDA,EXIT,SETRPR
   186
   187
   188 000100                                  STATE$  LFS,
   189 000100                                   TRAN$  "LF"
   190
   191 000100                                  STATE$
   192 000100                                   TRAN$  '.,$EXIT
   193
```

```
                                                                  F  7
UPDSLT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 3

MACRO CROSS REFERENCE                                    CREF    04.00

MACRO NAME        REFERENCES

DBGTP$            #5-141       #5-196       #5-203       #5-204       #5-205       #5-206       #5-207       #5-208       #5-209       #5-210
ISTAT$            #4-94         5-141
MTRAN$            #5-141
PRINT             #4-60
STATE$            #4-94         5-143       #5-146       #5-149       #5-152       #5-155       #5-158       #5-161       #5-164       #5-167
                  #5-170       #5-173       #5-176       #5-178       #5-180       #5-182       #5-184       #5-188       #5-191       #5-194
                  #5-198       #5-201       #5-214       #5-217
TRAN$             #4-94        #5-144       #5-147       #5-150       #5-153       #5-156       #5-159       #5-162       #5-165       #5-168
                  #5-171       #5-174       #5-177       #5-179       #5-181       #5-183       #5-185       #5-189       #5-192       #5-195
                  #5-196       #5-199       #5-202       #5-203       #5-204       #5-205       #5-206       #5-207       #5-208       #5-209
                  #5-210       #5-215
$PTNMB            #4-88
$PTNUM            #4-87         10-405       10-423       10-429       10-435
$$$PTN            #4-67         4-87         4-88
```

```
                                                                  G  7
**FILE**ID**UPDSTA
```

```
                                                                 F  8
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE                REFERENCES

CS.MOD    = ******  GX       6-185
CS.STA    = ******  GX       6-178
C.ADD       000005          #4-93      7-223
C.APR       000010          #4-96      9-265
C.CST       000007          #4-95      8-244
C.FLG       000006          #4-94      10-288    10-294    10-300    10-306
C.HTM       000012          #4-98      11-337
C.LEN     = 000016          #4-100
C.STA       000004          #4-92
C.STS     = ******  GX       6-178     6-185     11-335
F.NRBD    = ******  GX      *6-187
SENA        000002 R        #4-76      4-77      10-290
SENAL     = 000007          #4-77      10-291
SETADD      000150 R        #7-220
SETAPR      000226 R        #9-263
SETCST      000200 R        #8-242
SETFLG      000254 R        #10-284
SETHTM      000432 R        #11-333
SF.ENA    = ******  GX       10-288
SF.MFL    = ******  GX       10-300
SF.SER    = ******  GX       10-306
SF.SVC    = ******  GX       10-294
SMFL        000011 R        #4-78      4-79      10-302
SMFLL     = 000007          #4-79      10-303
SSER        000027 R        #4-82      4-83      10-308
SSERL     = 000007          #4-83      10-309
SSVC        000020 R        #4-80      4-81      10-296
SSVCL     = 000007          #4-81      10-297
STAKTB      000000 RG       #5-107     6-192
STASTB      000000 RG       #5-107     6-193
ST$HTM    = ******  GX       11-335
TEMP        000000 R        #4-75     *6-184     7-221     8-243     9-264     10-286    11-334
$ALPHA    = 000022          #5-107
$ANY      = 000020          #5-107
$BLANK    = 000006          #5-107
$BUFI     = ******  GX       6-187
$CNVD     = ******  GX       7-225     8-246     11-340
$CNVO     = ******  GX       9-267
$DIGIT    = 000024          #5-107
$DNUMB    = 000014          #5-107
$EOS      = 000012          #5-107
$ERROR    = ******  GX      *6-196
$EXIT     = 000000          #5-107
$FAIL     = 177777          #5-107
$GPRM     = ******           5-107
$LAMDA    = 000000          #5-107
$MOVST    = ******  GX       6-189
$MVASC    = ******  GX       10-292    10-298    10-304    10-310
$NUMBR    = 000002          #5-107
$OBUF     = ******  GX       7-224    *7-226     8-245    *8-247     9-266    *9-268     10-287    10-312   *10-316
                             11-338   *11-341

$OFDB     = ******  GX      *6-187
```

```
                                                                 G  8
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 2
```

```
                                                                          F  9
**FILE**ID**UPDUNT







UU      UU   PPPPPPPP     DDDDDDDD     UU      UU   NN      NN   TTTTTTTTTT
UU      UU   PPPPPPPP     DDDDDDDD     UU      UU   NN      NN   TTTTTTTTTT
UU      UU   PP      PP   DD      DD   UU      UU   NN      NN       TT
UU      UU   PP      PP   DD      DD   UU      UU   NN      NN       TT
UU      UU   PP      PP   DD      DD   UU      UU   NNNN    NN       TT
UU      UU   PP      PP   DD      DD   UU      UU   NNNN    NN       TT
UU      UU   PPPPPPPP     DD      DD   UU      UU   NN  NN  NN       TT
UU      UU   PPPPPPPP     DD      DD   UU      UU   NN  NN  NN       TT
UU      UU   PP           DD      DD   UU      UU   NN    NNNN       TT
UU      UU   PP           DD      DD   UU      UU   NN    NNNN       TT
UU      UU   PP           DD      DD   UU      UU   NN      NN       TT           ....
UU      UU   PP           DD      DD   UU      UU   NN      NN       TT           ....
UUUUUUUUUU   PP           DDDDDDDD     UUUUUUUUUU   NN      NN       TT           ....
UUUUUUUUUU   PP           DDDDDDDD     UUUUUUUUUU   NN      NN       TT           ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                          G  9
UPDUNT - CFE UPDATE UNIT DEFINI MACRO V05.03b Saturday 29-Jun-85 05:58
Table of contents
```

```
                                .SBTTL  SETPCH - SET EMULATOR CHARACTERISTICS
                        ;+
                        ;*** - SETPCH - SET EMULATOR CHARACTERISTICS
                        ;
                        ; INPUT:
                        ;       .TEMP = TEMPLATE ADDRESS
                        ;
                        ; OUTPUT:
                        ;
                        ;-

000314                  SETPCH: $PTNUM  #C.PCH,,#UN$PCH
000340  000207                  RETURN

        000001                  .END
```

```
                                                            F 11
UPDX2P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE              REFERENCES

CS.UNI  = ******  GX       6-196
C.BSZ     000014          #4-133       5-170
C.HBT     000022          #4-136       5-173
C.LEN   = 000024          #4-137
C.LIN     000004          #4-130       6-191       6-195       6-201
C.RTC     000016          #4-134       5-171
C.RTT     000020          #4-135       5-172
C.WSZ     000012          #4-132       5-169
F.NRBD  = ******  GX      *5-158      *5-176      *5-178
SETLIN    000200 R         5-168      #6-191
X2PDF     000000 R        #4-140       4-141       5-164
X2PLN   = 000010          #4-141       5-165
$BUFI   = ******  GX       5-178
$MVASC  = ******  GX       5-166
$OBUF   = ******  GX      *5-167       5-174       6-192      *6-193       6-198      *6-199
$OFDB   = ******  GX      *5-158      *5-176      *5-178
$PTNUM  = ******  GX       5-169       5-170       5-171       5-172       5-173       6-195       6-201
$PTR50  = ******  GX       6-191
$UPX2P    000010 RG       #5-157
$WDATA  = ******  GX       5-177
.BUFF   = ******  GX       5-158       5-163       5-175
.COMMA  = ******  GX      *5-162      *6-194      *6-200
.FLAGS  = ******  GX      *5-161       6-196
.TEMP   = ******  GX      *5-159
```

```
                                                            G 11
UPDX2P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58      PAGE 2
```

Table of contents

```
     4-   57    MACRO CALLS AND LOCAL DATA
     5-  148    $UPX3P - UPDATE X.25 LEVEL 3 CHARACTERISTICS
     93                                                 .SBTTL  Equated Symbols and Literals
     94
     95 000000                                          .PSECT  MLDDAT,D,RW,REL,OVR,LCL
     96
     97                                         ;       Equated Symbols
     98
     99
    100         000002                          EVTLUN  =       2                               ; Event logging LUN
    101         000002                          EVTEFL  =       2                               ; Event logging QIO event flag
    102
    103                                                 .IFT
    104
    105         000001                          FCSLUN  =       1                               ; Input file LUN
    106         000001                          FCSEFL  =       1                               ; Input file I/O event flag
    107         001000                          FCSBSZ  =       512.                            ; Output file buffer size in bytes
    108         000006                          MXFNLN  =       6                               ; Maximum microcode filename length
    109         014010                          CVTPRM  =       14010                           ; $CBTA conversion parameters
    110
    111                                                 .IFTF
    112                                                 .IF     DF      N$$UMR
    113
    114         000001                          LDBWTS  =       1                               ; Secs between LDB allocation attempts
    115         000003                          TMREFL  =       3                               ; Mark time event flag
    116
    117                                                 .ENDC
    118
    119                                                 .IFF
    120                                                 .IF     DF      N$$UMR
    121
    122                                         NBUFS   =       1                               ; Only use one CCB buffer descriptor
    123
    124                                                 .IFF
    125
    126                                         NBUFS   =       2                               ; Use both CCB buffer descriptors
    127
    128                                                 .ENDC
    129
    130
    131                                                 .IFTF
    132
    133
    134                                         ;       Event Logging Data
    135
    136
    137                                                 .IF     DF      P$$LUT
    138
    139                                         ETYPCL  =       CL$PAZ                          ; ASCII events are PLUTO-specific
    140
    141                                                 .IFF
    142
    143          010500                         ETYPCL  =       CL$ASZ                          ; ASCII events are RSX-specific
    144
    145                                                 .ENDC
    146
    147 000000        115       151       143   MLDERR: .ASCIZ  /Microcode load error on controller %R-%M/
    148 000051        110       141       162   DEVERR: .ASCIZ  /Hard device error/
    149
```

```
                                                   .SBTTL  FCSERR  - Input File FCS Error Handler
                                         ;+
                                         ;                                   *** FCSERR ***
                                         ;
                                         ; Function:       To log FCS errors as DECnet events.  No file handling is done
                                         ;                 to correct the error condition.
                                         ;
                                         ; Call Format:    CALL    FCSERR
                                         ;
                                         ; Inputs:         None
                                         ;
                                         ; Outputs:        R3 = load halt status
                                         ;                 FCS error is logged to event sink
                                         ;                 Current load sequence is marked for a halt
                                         ;
                                         ; Registers:      R0 through R4 are destroyed
                                         ;
                                         ;-

001302  012700  000144'                  FCSERR: MOV     #FDB,R0                 ; Get address of input file's FDB
001306  126027  000052  000000G                  CMPB    F.ERR(R0),#IE.EOF       ; Did end of file occur?
001314  001002                                   BNE     10$                     ; Branch if so, no real error
001316  005003                                   CLR     R3                      ; Get "end of data" halt reason
001320  000422                                   BR      20$                     ; Branch to common exit

001322  012702  001310'                  10$:    MOV     #FILERR,R2              ; Get $EDMSG argument block address
001326  116001  000052                           MOVB    F.ERR(R0),R1            ; Get FCS error code with sign extend
001332  010112                                   MOV     R1,(R2)                 ; Move error to argument block
001334  012700  001406'                          MOV     #WORKSP,R0              ; Get address of output string
001340  012701  000073'                          MOV     #FIOERR,R1              ; Get $EDMSG instruction string address
001344                                           CALL    $EDMSG                  ; Format the output string
001350  105010                                   CLRB    (R0)                    ; Make it an ASCIZ string
001352  012702  001406'                          MOV     #WORKSP,R2              ; Recover error-specific string address
001356                                           CALL    LOGEVT                  ; Log the event
001362  012703  001000                           MOV     #CS.FTL,R3              ; Get "fatal error" status

001366  000261                           20$:    SEC                             ; Set C-bit for error
001370                                           RETURN                          ; Exit
```

```
                                                                      F 15
MLD        CREATED BY  MACRO  ON 28-AUG-85 AT 11:27       PAGE 2
SYMBOL CROSS REFERENCE                                    CRFF    04.00

SYMBOL  VALUE                 REFERENCES

DL$SYN  = 000005              #6-83
DSDS      001312 R             8-170      #8-177      *9-299      *9-306      *14-756     *14-757     *14-758
DSPDRV    000604 R             9-466      #10-499      10-560      15-805
D.NAM     000004               9-287
D.UCB     000002               9-288
D.UCBL    000010               9-289
D.UNIT    000006               9-291
EF$ACT  = 000001              #6-83
ETYPCL  = 010500              #7-143       11-617
EVTBLK    001526 R            #8-216       11-606      11-615      11-617
EVTEFL  = 000002              #7-101       11-617
EVTLUN  = 000002              #7-100       9-269       11-617
EV$ACF  = 000201              #6-83
EV$ADR  = 000420              #6-83
EV$ADU  = 000417              #6-83
EV$APL  = 000400              #6-83
EV$ARC  = 000421              #6-83
EV$AUC  = 000010              #6-83
EV$AUS  = 000003              #6-83
EV$CDF  = 000520              #6-83
EV$COZ  = 000011              #6-83
EV$DBR  = 000302              #6-83
EV$GAS  = 035101              #6-83
EV$HCE  = 035114              #6-83
EV$HCI  = 035113              #6-83
EV$HFE  = 000506              #6-83
EV$IFL  = 000413              #6-83
EV$IFO  = 000415              #6-83
EV$IFS  = 000414              #6-83
EV$INF  = 000515              #6-83
EV$LDL  = 000407              #6-83
EV$LDN  = 010416              #6-83
EV$LDO  = 000411              #6-83
EV$LDS  = 000410              #6-83
EV$LSC  = 000500              #6-83
EV$LUP  = 000412              #6-83
EV$NOL  = 000402              #6-83
EV$NRC  = 000416              #6-83
EV$NSC  = 000200              #6-83
EV$NUL  = 000401              #6-83
EV$NVR  = 000406              #6-83
EV$OPL  = 000403              #6-83
EV$PCC  = 034000              #6-83
EV$PCI  = 034001              #6-83
EV$PCM  = 034002              #6-83
EV$PFE  = 000404              #6-83
EV$PPC  = 034003              #6-83
EV$RCF  = 000517              #6-83
EV$RDC  = 010001              #6-83
EV$RDR  = 010002              #6-83
EV$RJE  = 035106              #6-83
EV$RSC  = 000501              #6-83
```

```
                                                                      G 15
MLD        CREATED BY  MACRO  ON 28-AUG-85 AT 11:27       PAGE 3
```

```
     49                                         .SBTTL  Macro Definitions
     50
     51
     52                                 ;       System Macros Used
     53
     54                                         .IF     DF      R$$MPL
     55
     56                                         .MCALL  KRBDF$, RMAF$S, STAF$S, DIR$, GIN$
     57
     58                                         KRBDF$
     59
     60                                         .ENDC
     61
     62                                         .IF     NDF     R$$11S
     63
     64                                         .MCALL  FSRSZ$, OPEN$R, CLOSE$, READ$,  WAIT$,  FDBDF$
     65                                         .MCALL  FDAT$A, FDRC$A, FDBK$A, FDOP$A, FINIT$, UCBDF$, DCBDF$
     66
     67                                         UCBDF$
     68                                         DCBDF$
     69
     70                                         .IFF
     71
     72                                         .MCALL  PCBDF$
     73
     74 000000                                  PCBDF$
     75
     76                                         .IFTF
     77
     78                                         .MCALL  EXIT$S, SPRA$S, SREA$S, QIOW$S, ALUN$S
     79                                         .MCALL  TCBDF$, CCBDF$, EVLDF$, PDVDF$, SLTDF$
     80
     81 000000                                  TCBDF$
     82 000000                                  CCBDF$
     83 000000                                  EVLDF$
     84 000000                                  PDVDF$
     85 000000                                  SLTDF$
     86
     87                                         .IF     DF      N$$UMR
     88
     89                                         .MCALL  MRKT$,  STSE$,  DIR$
     90
     91                                         .ENDC
```

Table of contents

```
     4-   57     MACRO CALLS AND LOCAL DATA
     4-   58     MACRO DEFINITIONS
     5-  131     LOCAL DATA
     6-  144     $UPPVC - UPDATE PERMANENT VIRTUAL CIRCUIT
     7-  193     SETSTA - SET STATE
```

```
                        .SBTTL  MACRO CALLS AND LOCAL DATA
                        .SBTTL  MACRO DEFINITIONS
               ;
               ; LOCAL MACROS
               ;

                        .MACRO  PRINT   TEXT
                        MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                        CALL    $PRINT          ; PRINT MESSAGE
                        .ENDM   PRINT

                        .MACRO  ERROR$  TEXT
               ;
               ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
               ; MESSAGE STRING.
               ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
                        .ENDC
                        CALL    $CFERR                  ; PRINT ERROR MESSAGE
                        .ENDM   ERROR$

                        .MACRO  EPRINT  TEXT
               ;
               ; PRINT TEXT ON ERROR LUN
               ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ASCIZ STRING TO PRINT
                        .ENDC
                        CALL    $CFER1                  ; PRINT MESSAGE ON ERROR LUN
                        .ENDM   EPRINT
               ;
               ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
               ;
                        .MACRO  $$$PTN  ROU
                        .MACRO  'ROU    OFS,OCTL,OPT,?L
                        .IF NB <OPT>
                        MOV     OPT,R2
                        CALL    $PTOPT
                        BCS     L
                        .IFTF
                        MOV     OFS,R2
                        .IF NB <OCTL>
                        MOV     SP,R1
                        .IFF
                        CLR     R1
                        .ENDC ; IF NB <OCTL>
                        CALL    'ROU
                        .IFT  ; IF NB <OPT>
               L:
                        .ENDC ; IF NB <OPT>
                        .ENDM ; 'ROU
                        .ENDM ; $$$PTN

 000000                 $$$PTN  $PTNUM
 000000                 $$$PTN  $PTNMB
               ;
```

```
                                                .SBTTL  $UPREM - SETUP REMOTE DEFINITION
                                        ;+
                                        ; *** - $UPREM - SETUP REMOTE DEFINITION
                                        ;
                                        ; INPUT:
                                        ;       R3-R5   - TPARS REGISTERS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000154                                  $UPREM::
000154  012767  000033' 000000C                 MOV     #BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
000162  010546                                  MOV     R5,-(SP)        ; SAVE CURRENT TPARS STATE
000164  012700  000000G                 5$:     MOV     #$TMLST,R0      ; POINT AT START OF TEMPLATES
000170  010001                          10$:    MOV     R0,R1           ; SAVE ADDRESS OF CURRENT TEMPLATE
000172  011000                                  MOV     (R0),R0         ; GET ADDRESS OF NEXT TEMPLATE
000174  001455                                  BEQ     30$             ; IF EQ, END OF TEMPLATES
000176  122760  000000G 000000G                 CMPB    #CS.REM,C.STS(R0) ; IS THIS A REMOTE DEFINITION
000204  001371                                  BNE     10$             ; IF NE, NO .. KEEP LOOKING
                                        ;
                                        ; FOUND A TEMPLATE
                                        ;
000206  011011                                  MOV     (R0),(R1)       ; REMOVE TEMPLATE FROM CHAIN
000210  010067  177564                          MOV     R0,TEMP         ; SAVE TEMPLATE ADDRESS
000214  012700  000033'                         MOV     #BUFF,R0        ; POINT AT START OF OUTPUT BUFFER
000220  012701  000023'                         MOV     #REMDF,R1       ; POINT AT START OF REMOTE DEFINITION
000224  012702  000010                          MOV     #REMLN,R2       ; GET LENGTH OF REMOTE DEFINITION
000230  004767  000000G                         CALL    $MVASC          ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
000234  010067  000000G                         MOV     R0,$OBUF        ; SAVE CURRENT POSITION IN OUTPUT BUFFER
000240  005001                                  CLR     R1              ; IGNORE BLANKS
000242  012702  000000'                         MOV     #REMKTB,R2      ; KEYWORD TABLE
000246  012703  000020                          MOV     #REMLEN,R3      ; LENGTH OF REMOTE DEFINITION
000252  012704  000002'                         MOV     #REMDEF,R4      ; ADDRESS OF REMOTE DEFINITION
000256  012705  000000'                         MOV     #REMSTB,R5      ; STATE TABLE
000262  004767  000000G                         CALL    .TPARS          ; PARSE THE REST OF THE LINE
000266  103003                                  BCC     20$             ; NORMAL RETURN IF NO ERROR
000270  005267  000000G                         INC     $ERROR          ; INDICATE SYNTAX ERROR
000274  000415                                  BR      30$             ; AND LEAVE
000276  016701  000000G                 20$:    MOV     $OBUF,R1        ; CALCULATE LENGTH OF BUFFER
000302  162701  000033'                         SUB     #BUFF,R1        ;       ...
000306  010167  000000C                         MOV     R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
000312  004767  000000G                         CALL    $WDATA          ; WRITE THE RECORD TO THE OUTPUT FILE
000316  004767  000154                          CALL    UPDLL           ; SEE IF ANY DLL CHARACTERISTICS FOR THIS NODE
000322  103320                                  BCC     5$              ; IF CC, GO
000324  005267  000000G                         INC     $ERROR          ; SYNTAX ERROR SOMEWHERE
000330  012605                          30$:    MOV     (SP)+,R5        ; RESTORE R5
000332  012767  000000G 000000C                 MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
000340  000207                                  RETURN
```

```
                                  ;****
                                  ; INITIALIZE TABLE GENERATION
                                  ;****


 000002                                   ISTAT$  ROUSTB,ROUKTB


 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;MAX ADDRESS

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;COST

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;HOPS

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;ROUTING TIMER

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;NUMBER OF AREAS

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;BRO ENDNODE ADJACENCIES

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;AREA MAX HOPS

 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;AREA MAX COST
 000002                                   STATE$
 000002                                    TRAN$  <',>,,$MOVPT

 000002                                   STATE$
 000002                                    TRAN$  $NUMBR,,SETNUM                 ;LEVEL 2 TIMER
```

```
                                                        .SBTTL  PUTNOD - WRITE OUT NODE ADDRESS
                                                ;+
                                                ; *** - PUTNOD - WRITE OUT NODE ADDRESS
                                                ;
                                                ; CALLING SEQUENCE:
                                                ;
                                                        JSR     R5,PUTNOD       ; OUTPUT REMOTE NODE ADDRESS
                                                        .WORD   OFFSET,FLAG BIT ;  ...
                                                ;
                                                ; OUTPUT:
                                                        C-BIT = SUCCESS/FAILURE
                                                ;
                                                ;-
 000266                                         PUTNOD:
 000266  016700  000000G                                MOV     $OBUF,R0        ; POINT AT CURRENT POSITION IN LINE
 000272  016702  000000G                                MOV     .TEMP,R2        ; POINT AT TEMPLATE
 000276  062502                                         ADD     (R5)+,R2        ; POINT TO NODE ADDRESS IN TEMPLATE
 000300  112720  000054                                 MOVB    #<',>,(R0)+     ; ASSUME PARAMETER NOT PARSED
 000304  032577  000000G                                BIT     (R5)+,@.FLAGS   ; WAS PARAMETER PARSED ?
 000310  001421                                         BEQ     10$             ; IF EQ, NO
 000312  005300                                         DEC     R0              ; BACK OVER COMMA
 000314  011201                                         MOV     (R2),R1         ; GET NODE ADDRESS
 000316  010146                                         MOV     R1,-(SP)        ; COPY IY
 000320  042701  001777                                 BIC     #1777,R1        ; ONLY WANT AREA
 000324  000301                                         SWAB    R1              ; SHIFT IT INTO LOW BYTE
 000326  006201                                         ASR     R1              ; NOT DONE YET
 000330  006201                                         ASR     R1              ; ...
 000332  004767  000000G                                CALL    $CNVD           ; CONVERT NODE AREA TO DECIMAL
 000336  112720  000054                                 MOVB    #<',>,(R0)+     ; INSERT COMMA
 000342  012601                                         MOV     (SP)+,R1        ; RECOVER NODE ADDRESS
 000344  042701  176000                                 BIC     #^C1777,R1      ; CLEAR AREA
 000350  004767  000000G                                CALL    $CNVD           ; PUT LOW ORDER
 000354  112720  000054                         10$:    MOVB    #<',>,(R0)+     ; INSERT COMMA
 000360  010067  000000G                                MOV     R0,$OBUF        ; RESTORE BUFFER POINTER
 000364  000205                                         RTS     R5              ; DONE
```

```
    194 000100                          STATE$  SKIP
    195 000100                           TRAN$  '!',NEXT
    196 000100                           TRAN$  <',>,$EXIT
    197
    198 000100                          STATE$  NEXT
    199 000100                           TRAN$  !LFS
    200
    201 000100                          STATE$  BITS
    202 000100                           TRAN$  ''TIM'',SKIP
    203 000100                           TRAN$  ''MFL'',SKIP
    204 000100                           TRAN$  ''MDC'',SKIP
    205 000100                           TRAN$  ''MTP'',SKIP
    206 000100                           TRAN$  ''REA'',SKIP
    207 000100                           TRAN$  ''ENA'',SKIP
    208 000100                           TRAN$  ''X2P'',SKIP
    209 000100                           TRAN$  ''BRO'',SKIP
    210 000100                           TRAN$  ''SER'',SKIP
    211                         ;
    212                         ; EXIT STATE
    213                         ;
    214 000100                          STATE$  EXIT
    215 000100                           TRAN$  $LAMDA,$EXIT,$CLNUP
    216
    217 000100                          STATE$
```

```
**FILE**ID**UPDSTA




UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   TTTTTTTTTT     AAAAAA
UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   TTTTTTTTTT     AAAAAA
UU      UU   PP      PP   DD      DD   SS               TT       AA      AA
UU      UU   PP      PP   DD      DD   SS               TT       AA      AA
UU      UU   PP      PP   DD      DD   SS               TT       AA      AA
UU      UU   PP      PP   DD      DD   SS               TT       AA      AA
UU      UU   PPPPPPPP     DD      DD     SSSSSS         TT       AA      AA
UU      UU   PPPPPPPP     DD      DD     SSSSSS         TT       AA      AA
UU      UU   PP           DD      DD           SS       TT       AAAAAAAAAA
UU      UU   PP           DD      DD           SS       TT       AAAAAAAAAA
UU      UU   PP           DD      DD           SS       TT       AA      AA      ....
UU      UU   PP           DD      DD           SS       TT       AA      AA      ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS         TT       AA      AA      ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS         TT       AA      AA      ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

G 8

```
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE               REFERENCES

$RAD50  = 000016            #5-107
$RONLY  = ******             5-107        5-107        5-107
$STRNG  = 000004            #5-107
$SUBXP  = 000010            #5-107
$TMLST  = ******    GX       6-174
$UPSTA    000036 RG         #6-173
$$$FLG  = 177777            #5-107
$$$KEY  = 177777            #5-107
.CONER  = ******    GX       6-202
.TPARS  = ******    GX       6-194
```

H 8

```
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 3
```

Table of contents

```
     6-  153    $UPUNT - UPDATE UNIT DEFINITION
     7-  196    SETUNT - CHECK UNIT NUMBER
     8-  218    SETCW0 - SETUP THE CHARACTERISTICS WORD 0
     9-  239    SETCW1 - SET THE CHARACTERISTICS WORD 1
    10-  261    SETCSR - SET THE SECONDARY CSR ADDRESS
    11-  275    SETCST - SET THE LINE COST
    12-  289    SETDPR - SET THE LINE DEAD POLLING RATIO
    13-  303    SETPCH - SET EMULATOR CHARACTERISTICS
```

```
UPDUNT - CFE UPDATE UNIT DEFINI MACRO V05.03b Saturday 29-Jun-85 05:58  Page 13-1
Symbol table

CS.UNT= ****** GX       SETCST  000240R           UN$SCS= ****** GX    $FAIL = 177777        $TMLST= ****** GX
C.CSR   000012          SETCWO  000136R           $ALPHA= 000022       $LAMDA= 000000        $UPUNT  000000RG
C.CST   000014          SETCW1  000164R           $ANY  = 000020       $MOVPT= ****** GX     $$$FLG= 177777
C.CWO   000006          SETDPR  000266R           $BLANK= 000006       $MOVST= ****** GX     $$$KEY= 177777
C.CW1   000010          SETPCH  000314R           $CLNUP= ****** GX    $NUMBR= 000002        $$$STA= 000000
C.DPR   000016          SETUNT  000110R           $CNVO = ****** GX    $OBUF = ****** GX     .COMMA= ****** GX
C.LEN = 000026          UNTKTB  000000RG      003 $DIGIT= 000024       $PTNUM= ****** GX     .CONER= ****** GX
C.PCH   000024          UNTSTB  000000RG      002 $DNUMB= 000014       $PTOPT= ****** GX     .FLAGS= ****** GX
C.STS = ****** GX       UN$CST= ****** GX         $EOS  = 000012       $RAD50= 000016        .PNUMB= ****** GX
C.UNT   000004          UN$DPR= ****** GX         $ERROR= ****** GX    $STRNG= 000004        .TEMP = ****** GX
SETCSR  000212R         UN$PCH= ****** GX         $EXIT = 000000       $SUBXP= 000010        .TPARS= ****** GX


. ABS.  000026    000   (RW,I,GBL,ABS,OVR)
        000342    001   (RW,I,LCL,REL,CON)
$STATE  000046    002   (RW,D,LCL,REL,CON)
$KTAB   000000    003   (RW,D,LCL,REL,CON)
$KSTR   000000    004   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10796 Words  ( 43 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:18.33
SY:UPDUNT.V2,[132,134]UPDUNT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDUNT
```

```
UPDUNT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58     PAGE 1
```

```
UPDX2P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:58      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

EPRINT           #4-78
ERROR$           #4-67
PRINT            #4-62
$PTNMB           #4-111
$PTNUM           #4-110      5-169      5-170      5-171      5-172      5-173      6-195      6-201
$PTR50           #4-115      6-191
$$$PTN           #4-90       4-110      4-111
```

```
**FILE**ID**UPDX29
```

```
        .TITLE  UPDX3P - UPDATE X.25 LEVEL 3 CHARACTERISTICS
        .IDENT  /V05.00/
        .ENABL  LC

;
; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       UPDATE X.25 LEVEL 3 CHARACTERISTICS IN CETAB
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-AUG-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;



```

```
    150                                         .IFT
    151
    152 000073     106     151      154  FIOERR: .ASCIZ  /File access error = %D, file = %VA%VA%VA/
    153
    154                                         .IFF
    155
    156                                  PARERR: .ASCIZ  /Partition %2P not found/
    157
    158                                         .IFTF
    159                                         .EVEN
```

```
                                            .SBTTL  LOCFIL  - Locate Microcode File
                                    ;+
                                    ;                                *** LOCFIL ***
                                    ;
                                    ; Function:     To locate a microcode task image file on the current network
                                    ;               UIC of MLD's load device.  If the file is found, it is opened
                                    ;               for read with block I/O.
                                    ;
                                    ; Call Format:  CALL    LOCFIL
                                    ;
                                    ; Inputs:       R4 = address of filename/type/version ASCII string
                                    ;               R5 = length of the string
                                    ;
                                    ; Outputs:      C-bit clear = file found and opened
                                    ;               C-bit set   = file not found or could not be opened
                                    ;
                                    ;               BLOCK(+2) is set up for first file read
                                    ;
                                    ; Registers:    R0 through R5 destroyed
                                    ;
                                    ;-

001372  020527  000006              LOCFIL: CMP     R5,#MXFNLN              ; Does filename length exceed maximum?
001376  003402                              BLE     10$                     ; Branch if not
001400  012705  000006                      MOV     #MXFNLN,R5              ; Else, force length to maximum

001404  010467  001324'             10$:    MOV     R4,DSDS+10.             ; Move address of dataset descriptor
001410  010567  001322'                     MOV     R5,DSDS+8.              ; Move string length as well
001414  062767  000004  001322'             ADD     #4,DSDS+8.              ; Add in length of ".TSK" extension
001422  060504                              ADD     R5,R4                   ; Point past end of string
001424  112724  000056                      MOVB    #'.,(R4)+               ; Append the string ".TSK"
001430  112724  000124                      MOVB    #'T,(R4)+               ;   to the end of the specified
001434  112724  000123                      MOVB    #'S,(R4)+               ;     filename string
001440  112714  000113                      MOVB    #'K,(R4)                ; ...

001444                                      OPEN$R  #FDB,,,,,,FCSERR        ; Open the input file
001470  103405                              BCS     20$                     ; Branch if error occurred
001472  005067  001372'                     CLR     BLOCK                   ; Clear high virtual block number (VBN)
001476  012767  000002  001374'             MOV     #2,BLOCK+2              ; Set low VBN for first data block read

001504                              20$:    RETURN                          ; Exit
```

```
                                                          G 15
MLD       CREATED BY  MACRO  ON 28-AUG-85 AT 11:27     PAGE 3
SYMBOL CROSS REFERENCE                                 CREF   04.00

SYMBOL  VALUE              REFERENCES

EV$RUL  = 000405           #6-83
EV$SNA  = 035000           #6-83
EV$SNF  = 000516           #6-83
EV$SPE  = 035001           #6-83
EV$XCE  = 034110           #6-83
EV$XDI  = 013600           #6-83
EV$XGW  = 034111           #6-83
EV$XMX  = 000514           #6-83
EV$XRS  = 000512           #6-83
EV$XSC  = 000513           #6-83
EV$X2S  = 013500           #6-83
EV.CCB  = 000001           #6-83
EV.CIR  = 000020           #6-83
EV.LCB  = 000100           #6-83
EV.LIN  = 000004           #6-83
EV.MAP  = 000002           #6-83
EV.MOD  = 000040           #6-83
EV.NOD  = 000010           #6-83
EV.PRT  = 000200           #6-83
EXEVCL  = 000017           #8-247
EXEVEC    000000 R         #8-227      8-247
E$DATA    000020           #6-83       11-606
E$EVTS    000000           #6-83
E$LCN     000016           #6-83
E$LIN     000000           #6-83
E$MOD     000012           #6-83
E$NOD     000010           #6-83
E$PORT    000014           #6-83
E$PRM     000002           #6-83
E$STAT    000006           #6-83
E$TCB     000004           #6-83
E.CTL     000020           #6-83
E.DATA    000046           #6-83
E.EVT     000002           #6-83
E.LCN     000042           #6-83
E.LFN     000216           #6-83
E.LIN     000024           #6-83
E.LNK     000000           #6-83
E.MOD     000036           #6-83
E.NOD     000034           #6-83
E.PDV     000021           #6-83
E.PORT    000040           #6-83
E.PRM     000026           #6-83
E.PVC     000044           #6-83
E.SIZ     000022           #6-83
E.TIME    000004           #6-83
FCSBSZ  = 001000           #7-107      8-169       8-174
FCSBUF    000304 R          8-169     #8-174       15-801
FCSEFL  = 000001           #7-106      8-169
FCSERR    001302 R         #13-710     14-765      14-765      15-794      15-794
FCSIOS    001304 R          8-169     #8-175       15-803
FCSLUN  = 000001           #7-105      8-170
```

```
                                                          H 15
MLD       CREATED BY  MACRO  ON 28-AUG-85 AT 11:27     PAGE 4
```

```
                                        .SBTTL  Equated Symbols and Literals

000000                                  .PSECT  MLDDAT,D,RW,REL,OVR,LCL

                                ;       Equated Symbols


        000002                  EVTLUN  =       2                       ; Event logging LUN
        000002                  EVTEFL  =       2                       ; Event logging QIO event flag

                                        .IFT

                                FCSLUN  =       1                       ; Input file LUN
                                FCSEFL  =       1                       ; Input file I/O event flag
                                FCSBSZ  =       512.                    ; Output file buffer size in bytes
                                MXFNLN  =       6                       ; Maximum microcode filename length
                                CVTPRM  =       14010                   ; $CBTA conversion parameters

                                        .IFTF
                                        .IF     DF      N$$UMR

        000001                  LDBWTS  =       1                       ; Secs between LDB allocation attempts
        000003                  TMREFL  =       3                       ; Mark time event flag

                                        .ENDC

                                        .IFF
                                        .IF     DF      N$$UMR

        000001                  NBUFS   =       1                       ; Only use one CCB buffer descriptor

                                        .IFF

                                NBUFS   =       2                       ; Use both CCB buffer descriptors

                                        .ENDC


                                        .IFTF


                                ;       Event Logging Data


                                        .IF     DF      P$$LUT

                                ETYPCL  =       CL$PAZ                  ; ASCII events are PLUTO-specific

                                        .IFF

        010500                  ETYPCL  =       CL$ASZ                  ; ASCII events are RSX-specific

                                        .ENDC

000000     115     151     143  MLDERR: .ASCIZ  /Microcode load error on controller %R-%M/
000051     110     141     162  DEVERR: .ASCIZ  /Hard device error/

```

```
	.TITLE	UPDPVC - UPDATE PERMANENT VIRTUAL CIRCUIT
	.IDENT	/V05.00/
	.ENABL	LC

;
; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;	UPDATE PERMANENT VIRTUAL CIRCUIT
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	14-AUG-81
;	DECNET-11M/S V3.1
;	DECNET-11M-PLUS V1.1
;
; 3.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;



```

```
; PUT RAD50 TEMPLATE FIELD TO BUFFER
;
        .MACRO  $PTR50  OFS,WC,OPT,?L
        .IF NB <OPT>
        MOV     OPT,R2
        CALL    $PTOPT
        BCS     L
        .IFTF
        MOV     OFS,R2
        MOV     WC,R1
        CALL    $PTR50
        .IFT
L:
        .ENDC
        .ENDM
```

```
                                                .SBTTL  SETNAM - WRITE OUT REMOTE NODE NAME
                                        ;+
                                        ; *** - SETNAM - WRITE OUT REMOTE NODE NAME
                                        ;
                                        ; INPUT:
                                        ;       .PSTPT = THE ADDRESS OF THE NODE NAME STRING
                                        ;       .PSTCN = THE LENGTH OF THE NODE NAME STRING
                                        ;       $OBUF = CURRENT POSITION IN OUTPUT LINE
                                        ;       TEMP = TEMPLATE ADDRESS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000342                                  SETNAM:
000342  016700  000000G                         MOV     $OBUF,R0        ; POINT AT CURRENT POSITION IN LINE
000346  016702  177426                          MOV     TEMP,R2         ; POINT AT TEMPLATE
000352  016201  000004                          MOV     C.NAM(R2),R1    ; GET THE 1ST HALF OF THE NODE NAME
000356  004767  000000G                         CALL    $C5TA           ; CONVERT RAD50 TO ASCII
000362  016702  177412                          MOV     TEMP,R2         ; POINT AT TEMPLATE
000366  016201  000006                          MOV     C.NAM+2(R2),R1  ; GET THE 2ND HALF OF THE NODE NAME
000372  004767  000000G                         CALL    $C5TA           ; CONVERT RAD50 TO ASCII
000376  010067  000000G                         MOV     R0,$OBUF        ; UPDATE POINTER INTO OUTPUT LINE
000402  000207                                  RETURN
```

```
    152                                 ;
    153                                 ; EXIT STATE
    154                                 ;
    155 000002                                  STATE$  EXIT
    156 000002                                   TRAN$  $LAMDA,$EXIT,$CLNUP
    157
    158 000002                                  STATE$
```

```
                                        .SBTTL  PUTR50 - OUTPUT EXTENDED RAD50 STRING
                                ;+
                                ; *** PUTR50 - OUTPUT EXTENDED RAD50 STRING
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,PUTR50
                                ;       .WORD   TEMPLATE OFFSET
                                ;       .WORD   INPUT WORD COUNT
                                ;       .WORD   FLAG BIT
                                ;-

000366                          PUTR50:
000366                                  SAVRG   <R3,R4>         ; SAVE TPARS REGISTER
000372  016704  000000G                 MOV     .TEMP,R4        ; POINT TO TEMPLATE
000376  062504                          ADD     (R5)+,R4        ; OFFSET INTO IT ...
000400  012503                          MOV     (R5)+,R3        ; PICK UP INPUT WORD COUNT
000402  016700  000000G                 MOV     $OBUF,R0        ; SET UP BUFFER POINTER
000406  032577  000000G                 BIT     (R5)+,@.FLAGS   ; WAS THIS PARAMETER PARSED ?
000412  001404                          BEQ     20$             ; IF EQ, NO - DONE
000414  012401                  10$:    MOV     (R4)+,R1        ; PICK UP A WORD TO CONVERT
000416  004767  000000G                 CALL    $C5TA1          ; CONVERT IT
000422  077304                          SOB     R3,10$          ; TIL DONE
000424  112720  000054          20$:    MOVB    #<',>,(R0)+     ; INSERT COMMA
000430  010067  000000G                 MOV     R0,$OBUF        ; RESTORE BUFFER POINTER
000434                                  RESRG   <R4,R3>         ; RESTORE TPARS REGISTER
000440  000205                          RTS     R5              ; DONE
```

```
                                                .SBTTL  $UPSLT - UPDATE SLT DEFINITION
                                        ;+
                                        ; *** - $UPSLT - UPDATE SLT DEFINITION
                                        ;
                                        ; INPUT:
                                        ;       R3-R5   - TPARS REGISTERS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-
000100                                  $UPSLT::
000100  012700  000000G                         MOV     #$TMLST,R0      ; POINT AT START OF TEMPLATES
000104  010001                          10$:    MOV     R0,R1           ; SAVE CURRENT TEMPLATE
000106  011000                                  MOV     (R0),R0         ; GET NEXT TEMPLATE
000110  001433                                  BEQ     101$            ; IF EQ, END OF TEMPLATES
000112  122760  000000G 000000G                 CMPB    #CS.SLT,C.STS(R0) ; IS THIS AN SLT TEMPLATE ?
000120  001371                                  BNE     10$             ; IF NE, NO .. KEEP LOOKING
                                        ;
                                        ; ASSUME THIS IS THE RIGHT TEMPLATE
                                        ;
000122  011011                                  MOV     (R0),(R1)       ; REMOVE TEMPLATE FROM CHAIN
000124  010067  000000G                         MOV     R0,.TEMP        ; SAVE TEMPLATE ADDRESS
000130  005720                                  TST     (R0)+           ; POINT TO C.STS
000132  010067  000000G                         MOV     R0,.FLAGS       ; SAVE POINTER
000136  005067  000000G                         CLR     .COMMA          ; INITIALIZE COUNTER
000142  004767  000000G                         CALL    $MOVST          ; MOVE FIRST FIELD INTO OUTPUT BUFFER
000146  010546                                  MOV     R5,-(SP)        ; SAVE R5
000150  005001                                  CLR     R1              ; IGNORE BLANKS
000152  012702  000000'                         MOV     #SLTKTB,R2      ; KEYWORD TABLE
000156  012705  000000'                         MOV     #SLTSTB,R5      ; STATE TABLE
000162  004767  000000G                         CALL    .TPARS          ; PARSE THE REST OF THE LINE
000166  103002                                  BCC     20$             ; NORMAL RETURN IF NO ERROR
000170  005267  000000G                         INC     $ERROR          ; INDICATE SYNTAX ERROR
000174  012605                          20$:    MOV     (SP)+,R5        ; RESTORE R5
000176  000207                          30$:    RETURN
                                        ;
                                        ; ERROR CONDITIONS
                                        ;
000200  000167  000000G                 101$:   JMP     .CONER          ; CONSISTENCY ERROR
```

Table of contents

```
     6-  161     $UPSTA - UPDATE STATION DEFINITION
     7-  204     SETADD - SET UP STATION ADDRESS
     8-  229     SETCST - SET THE STATION COST
     9-  250     SETAPR - SET THE STATION ACTIVE POLLING RATIO
    10-  271     SETFLG - SET STATION FLAGS
    11-  320     SETHTM - SET THE HELLO TIMER
```

```
UPDSTA     CREATED BY  MACRO  ON 29-JUN-85 AT 05:57      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME         REFERENCES

DBGTP$             #5-107    #5-120    #5-124    #5-125    #5-132    #5-154
ISTAT$             #4-68      5-107
MTRAN$             #5-107
PRINT              #4-59
STATE$             #4-68      5-109    #5-112    #5-115    #5-118    #5-122    #5-127    #5-130    #5-134    #5-137
                   #5-140    #5-146    #5-149    #5-152    #5-156    #5-159
 TRAN$             #4-68     #5-110    #5-113    #5-116    #5-119    #5-120    #5-123    #5-124    #5-125    #5-128
                   #5-131    #5-132    #5-135    #5-138    #5-141    #5-147    #5-150    #5-153    #5-154    #5-157
```

```
**FILE**ID**UPDSVC
```

```
                                .TITLE  UPDUNT - CFE UPDATE UNIT DEFINITION IN CETAB
                                .IDENT  /V05.00/

                        ;
                        ; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985  BY
                        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                        ;
                        ;
                        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                        ; TRANSFERRED.
                        ;
                        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                        ; CORPORATION.
                        ;
                        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                        ;
                        ;
                        ; MODULE DESCRIPTION:
                        ;
                        ;       STATE TABLE TO PARSE THE UNIT DEFINITIONS
                        ;
                        ;
                        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                        ;
                        ; IDENT HISTORY:
                        ;
                        ; 1.00  14-DEC-79
                        ;       DECNET-11M/S V3.0
                        ;       DECNET-11M-PLUS V1.0
                        ;
                        ; 3.00  16-APR-82
                        ;       DECNET-11M V3.1
                        ;       DECNET-11M-PLUS V1.1
                        ;
                        ; 4.00  07-NOV-83
                        ;       DECNET-11M V4.0
                        ;       DECNET-11M-PLUS V2.0
                        ;
                        ; 5.00  22-JUL-85
                        ;       DECnet-11M/S V4.2
                        ;       DECnet-11M-Plus V3.0
                        ;       DECnet-Micro/RSX V1.0
                        ;
                        ;
```

```
                                                                      H 10
UPDUNT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:58        PAGE 1
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL    VALUE                 REFERENCES

CS.UNT    = ******   GX          6-170
C.CSR       000012              #4-110      10-272
C.CST       000014              #4-111      11-286
C.CWO       000006              #4-108      8-233
C.CW1       000010              #4-109      9-255
C.DPR       000016              #4-112      12-300
C.LEN     = 000026              #4-116
C.PCH       000024              #4-115      13-314
C.STS     = ******   GX          6-170
C.UNT       000004              #4-107      7-213
SETCSR      000212 R            #10-272
SETCST      000240 R            #11-286
SETCWO      000136 R            #8-231
SETCW1      000164 R            #9-253
SETDPR      000266 R            #12-300
SETPCH      000314 R            #13-314
SETUNT      000110 R            #7-210
UNTKTB      000000 RG           #5-124      6-183
UNTSTB      000000 RG           #5-124      6-184
UN$CST    = ******   GX          11-286
UN$DPR    = ******   GX          12-300
UN$PCH    = ******   GX          13-314
UN$SCS    = ******   GX          10-272
$ALPHA    = 000022              #5-124
$ANY      = 000020              #5-124
$BLANK    = 000006              #5-124
$CLNUP    = ******   GX          6-189
$CNVO     = ******   GX          8-235      9-257
$DIGIT    = 000024              #5-124
$DNUMB    = 000014              #5-124
$EOS      = 000012              #5-124
$ERROR    = ******   GX         *6-187
$EXIT     = 000000              #5-124
$FAIL     = 177777              #5-124
$GPRM     = ******               5-124
$LAMDA    = 000000              #5-124
$MOVPT    = ******   GX          7-211
$MOVST    = ******   GX          6-179
$NUMBR    = 000002              #5-124
$OBUF     = ******   GX          8-234     *8-236       9-256      *9-258
$PTNUM    = ******   GX          10-272     11-286      12-300      13-314
$PTOPT    = ******   GX          10-272     11-286      12-300      13-314
$RAD50    = 000016              #5-124
$RONLY    = ******               5-124      5-124       5-124
$STRNG    = 000004              #5-124
$SUBXP    = 000010              #5-124
$TMLST    = ******   GX          6-166
$UPUNT      000000 RG           #6-165
$$$FLG    = 177777              #5-124
$$$KEY    = 177777              #5-124
.COMMA    = ******   GX         *6-181
.CONER    = ******   GX          6-194
```

```
                                                                      I 10
UPDUNT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:58        PAGE 2
```

```
**FILE**ID**UPDX29








UU      UU   PPPPPPPP     DDDDDDDD     XX      XX     222222       999999
UU      UU   PPPPPPPP     DDDDDDDD     XX      XX     222222       999999
UU      UU   PP      PP   DD      DD   XX      XX   22      22   99      99
UU      UU   PP      PP   DD      DD   XX      XX   22      22   99      99
UU      UU   PP      PP   DD      DD     XX  XX             22   99      99
UU      UU   PP      PP   DD      DD     XX  XX             22   99      99
UU      UU   PPPPPPPP     DD      DD       XX           22       99999999
UU      UU   PPPPPPPP     DD      DD       XX           22       99999999
UU      UU   PP           DD      DD     XX  XX         22             99
UU      UU   PP           DD      DD     XX  XX         22             99
UU      UU   PP           DD      DD   XX      XX     22                 99       ....
UU      UU   PP           DD      DD   XX      XX     22                 99       ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX   2222222222     999999         ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX   2222222222     999999         ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                        .SBTTL  MACRO CALLS AND LOCAL DATA
                ;
                ; LOCAL MACROS
                ;

                        .MACRO  PRINT   TEXT
                        MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                        CALL    $PRINT          ; PRINT MESSAGE
                        .ENDM   PRINT

                        .MACRO  ERROR$  TEXT
                ;
                ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
                ; MESSAGE STRING.
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
                        .ENDC
                        CALL    $CFERR                  ; PRINT ERROR MESSAGE
                        .ENDM   ERROR$

                        .MACRO  EPRINT  TEXT
                ;
                ; PRINT TEXT ON ERROR LUN
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ASCIZ STRING TO PRINT
                        .ENDC
                        CALL    $CFER1                  ; PRINT MESSAGE ON ERROR LUN
                        .ENDM   EPRINT
                ;
                ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
                ;
                        .MACRO  $$$PTN  ROU
                        .MACRO  'ROU    OFS,OCTL,OPT,?L
                        .IF NB <OPT>
                        MOV     OPT,R2
                        CALL    $PTOPT
                        BCS     L
                        .IFTF
                        MOV     OFS,R2
                        .IF NB <OCTL>
                        MOV     SP,R1
                        .IFF
                        CLR     R1
                        .ENDC ; IF NB <OCTL>
                        CALL    'ROU
                        .IFT  ; IF NB <OPT>
                L:
                        .ENDC ; IF NB <OPT>
                        .ENDM ; 'ROU
                        .ENDM ; $$$PTN

 000000                 $$$PTN  $PTNUM
 000000                 $$$PTN  $PTNMB
                ;
                ; PUT RAD50 TEMPLATE FIELD TO BUFFER
```

```
                                         .SBTTL  Load Control Data
                                         .IFT

                               ;         Input File Descriptor Block (FDB) Definitions

    000144                     FDB:      FDBDF$                                             ; Allocate an FDB block
    000304                               FDAT$A                                             ; File already exists
    000304                               FDRC$A  FD.RWM                                     ; Read/write block I/O access
    000304                               FDBK$A  FCSBUF,FCSBSZ,,FCSEFL,FCSIOS               ; Set buffer/size, flag, status
    000304                               FDOP$A  FCSLUN,DSDS                                ; Specify LUN and filespec area

                               ;         General Data Declarations

    000304                     FCSBUF:  .BLKB    FCSBSZ                         ; Output file I/O buffer
    001304                     FCSIOS:  .BLKW    2                              ; FCS I/O status doubleword
    001310                     FILERR:  .BLKW    1                              ; FCS error storage for $EDMSG
    001312  000000             DSDS:    .WORD    0                              ; Length of source device name
    001314  001326'                     .WORD    FILDEV                         ; Source device name address
    001316  000000                      .WORD    0                              ; Length of source directory
    001320  001334'                     .WORD    FILDIR                         ; Source directory address
    001322  000000                      .WORD    0                              ; Length of source filename
    001324  000000                      .WORD    0                              ; Address of source filename
    001326                     FILDEV:  .BLKB    6                              ; Source device
    001334                     FILDIR:  .BLKB    10.                            ; Source directory

                                        .IF      DF R$$MPL
                                        .IF      NDF R$$PRO

                               GINVEC: GIN$      GI.VEC,EXEVEC,EXEVCL

                                        .ENDC    ; NDF R$$PRO
                                        .ENDC    ; DF R$$MPL

                                        .IFF

                               PCBADR: .BLKW     1                              ; Microcode PCB address

                                        .IFTF

                                        .IF      DF       N$$UMR

     001346                    SETTMR: MRKT$     TMREFL,LDBWTS,2                ; DPB for LDB wait timer
     001360                    WAIT:   STSE$     TMREFL                         ; DPB to stop task for TMREFL
     001364                    LDBPTR: .BLKW     1                              ; Address of CCB/LDB
     001366                    LDBIAS: .BLKW     1                              ; Mapping bias of LDB
     001370                    LDBADR: .BLKW     1                              ; Starting address of LDB

                                        .ENDC

     001372  000000  000000    BLOCK:   .WORD    0,0                            ; Block number for load control
     001376                    CURQST:  .BLKW    1                              ; Address of current load CCB
     001400                    PDVIDX:  .BLKW    1                              ; PDV index of device driver
     001402                    CTLNUM:  .BLKW    1                              ; Device controller number
     001404                    DDMTBL:  .BLKW    1                              ; Driver's line table address
     001406                    WORKSP:  .BLKB    80.                            ; Work space
     001526  000000  000000  000000  EVTBLK:  .WORD    0,0,0,2,0,0,0,0          ; Event descriptor block
     001546                             .BLKB    160.                           ; Event data area
```

```
                                        .SBTTL  MLOAD   - Load Control for RSX-11M Systems
                                ;+
                                ;                               *** MLOAD ***
                                ;
                                ; Function:     To control microcode load sequences on RSX-11M systems.
                                ;               This routine performs file I/O and maintains the load seq-
                                ;               uence counter(s) and status.
                                ;
                                ; Call Format:  CALL    MLOAD
                                ;
                                ; Inputs:       Input file is assumed to have been opened successfully.
                                ;               R4 = address of "load request" CCB
                                ;
                                ; Outputs:      Input file is closed, load sequence ended
                                ;               R3 = load halt status
                                ;               R4 = address of "halt load" CCB
                                ;
                                ; Registers:    R2, R3, and R5 are destroyed
                                ;
                                ;-

 001506  005267  001374'        MLOAD:  INC     BLOCK+2                 ; Update virtual block number
 001512                                 READ$   #FDB,,,#BLOCK,,,,FCSERR ; Read the next block in file
 001552  103423                         BCS     10$                     ; Branch if read could not be started
 001554                                 WAIT$   #FDB                    ; Wait for read to complete
 001564  103416                         BCS     10$                     ; Branch if disk error

 001566  010403                         MOV     R4,R3                   ; Copy CCB address
 001570  062703  000016                 ADD     #C.BUF1+2,R3            ; Point at first CCB buffer address
 001574  012713  000304'                MOV     #FCSBUF,(R3)            ; Copy buffer's APR5-biased address
 001600  062723  020000                 ADD     #20000,(R3)+            ; Force address to be APR6-biased
 001604  016723  001306'                MOV     FCSIOS+2,(R3)+          ; Get buffer byte count from I/O status
 001610  012713  100000                 MOV     #CF.LB,(R3)             ; One buffer/CCB on RSX-11M systems
 001614                                 CALL    DSPDRV                  ; Call driver to load the microcode
 001620  103332                         BCC     MLOAD                   ; Branch if load sequence not finished

 001622                         10$:    CLOSE$  #FDB                    ; Close the microcode file
 001632                                 RETURN                          ; Exit

                                        .IFF                            ; If system is RSX-11S ...
```

```
                                                                  H 15
MLD         CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

FC.MLD  = 000026             9-335      9-364      9-447      9-463      12-678
FDB       000144 R          #8-166     12-651     12-651     13-710      14-765     14-765     15-794     15-794     15-796
                             15-796    15-808     15-808
FD.RWM  = ******   GX        8-168
FILDEV    001326 R           8-178     #8-183     *9-287      9-294       9-298
FILDIR    001334 R           8-180     #8-184      9-301      9-305
FILERR    001310 R          #8-176     13-716
FIOERR    000073 R          #7-152     13-720
FO.RD   = ******   GX        14-765
FR$BCC  = 000007            #6-83
FR$CCF  = 000001            #6-83
FR$CDF  = 000002            #6-83
FR$DAO  = 000011            #6-83
FR$EXC  = 000000            #6-83
FR$FRM  = 000010            #6-83
FR$FTL  = 000005            #6-83
FR$OPN  = 000004            #6-83
FR$RFD  = 000006            #6-83
FR$SBU  = 000012            #6-83
FR$SHO  = 000003            #6-83
FR$UBU  = 000013            #6-83
FR$UPT  = 000014            #6-83
FS.HLT  = 000000             9-463     12-678
FS.RLB  = 002000             9-364
FS.WLB  = 001000             9-335      9-447     10-508
F.BKDS  = 000020             8-169      8-169
F.BKEF  = 000050             8-169
F.BKST  = 000024             8-169
F.BKVB  = 000064            *15-794    *15-794
F.DSPT  = 000044             8-170
F.ERR   = 000052             13-711    13-717
F.FACC  = 000043            *14-765
F.LUN   = 000042             8-170
F.RACC  = 000016             8-168
IE.EOF  = ******   GX        13-711
IO.WLB  = ******   GX        11-617
ISAR5     000002 RG         #8-229     10-513
I$$AS   = ******             6-84
KISAR5  = ******   GX        8-229
LDBADR    001370 R          #8-206     *9-334     10-528     10-532
LDBGT     000020 RG         #8-239      9-330
LDBIAS    001366 R          #8-205     *9-333     10-527
LDBPTR    001364 R          #8-204     *9-314     *9-332      9-338       9-371     *9-373      9-394     10-530     10-555
                             12-659
LDBRT     000022 RG         #8-240      9-372     12-661
LDBWTS  = 000001            #7-114      8-202
LF.ACT  = 100000            #6-85
LF.BRO  = 000400            #6-85
LF.BWT  = 000007            #6-85
LF.ENA  = 002000            #6-85
LF.LPB  = 001000            #6-85
LF.MDC  = 000100            #6-85
```

```
                                                                  I 15
MLD         CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 5
```

```
    150                                                  .IFT
    151
    152                                          FIOERR: .ASCIZ  /File access error = %D, file = %VA%VA%VA/
    153
    154                                                  .IFF
    155
    156 000073      120      141      162        PARERR: .ASCIZ  /Partition %2R not found/
    157
    158                                                  .IFTF
    159                                                  .EVEN
```

```
     57                                 .SBTTL  MACRO CALLS AND LOCAL DATA
     58                                 .SBTTL  MACRO DEFINITIONS
     59                         ;
     60                         ; LOCAL MACROS
     61                         ;
     62
     63                                 .MACRO  PRINT   TEXT
     64                                 MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
     65                                 CALL    $PRINT          ; PRINT MESSAGE
     66                                 .ENDM   PRINT
     67
     68                                 .MACRO  ERROR$  TEXT
     69                         ;
     70                         ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
     71                         ; MESSAGE STRING.
     72                         ;
     73                                 .IF DIF <TEXT><R0>
     74                                 MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
     75                                 .ENDC
     76                                 CALL    $CFERR                  ; PRINT ERROR MESSAGE
     77                                 .ENDM   ERROR$
     78
     79                                 .MACRO  EPRINT  TEXT
     80                         ;
     81                         ; PRINT TEXT ON ERROR LUN
     82                         ;
     83                                 .IF DIF <TEXT><R0>
     84                                 MOV     TEXT,R0                 ; GET ADDRESS OF ASCIZ STRING TO PRINT
     85                                 .ENDC
     86                                 CALL    $CFER1                  ; PRINT MESSAGE ON ERROR LUN
     87                                 .ENDM   EPRINT
     88                         ;
     89                         ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
     90                         ;
     91                                 .MACRO  $$$PTN  ROU
     92                                 .MACRO  'ROU    OFS,OCTL,OPT,?L
     93                                 .IF NB <OPT>
     94                                 MOV     OPT,R2
     95                                 CALL    $PTOPT
     96                                 BCS     L
     97                                 .IFTF
     98                                 MOV     OFS,R2
     99                                 .IF NB <OCTL>
    100                                 MOV     SP,R1
    101                                 .IFF
    102                                 CLR     R1
    103                                 .ENDC ; IF NB <OCTL>
    104                                 CALL    'ROU
    105                                 .IFT  ; IF NB <OPT>
    106                         L:
    107                                 .ENDC ; IF NB <OPT>
    108                                 .ENDM ; 'ROU
    109                                 .ENDM ; $$$PTN
    110
    111 000000                          $$$PTN  $PTNUM
    112 000000                          $$$PTN  $PTNMB
    113                         ;
    130                                                    .SBTTL  LOCAL DATA
    131
    132                                         ;
    133                                         ; DEFINE THE REMOTE DTE BLOCK
    134                                         ;
    135 000000                                          .ASECT
    136         000000                                  .=0
    137 000000                                          .BLKW   1                 ; C.LNK
    138 000002                                          .BLKW   1                 ; C.STS
    139 000004                                 C.NAM:   .BLKW   2.                ; RAD50 DTE NAME
    140 000010                                 C.ADR:   .BLKW   5.                ; RAD50 DTE ADDRESS (1.-15. DECIMAL DIGITS)
    141         000022                         C.LEN=.                            ; LENGTH OF THE BLOCK
    142 000000                                          .PSECT
    143                                                 .NLIST  BEX
    144 000000     011      122      104       RDTDF:   .ASCII  <11>/RDT$DF/<11>
    145         000010                                  RDTLN=.-RDTDF
    146                                                 .LIST   BEX
    147                                                 .EVEN
    148
```

```
                                        .SBTTL  SETARE - WRITE OUT AREA
                                ;+
                                ;  *** - SETADD - WRITE OUT AREA
                                ;
                                ;  INPUT:
                                ;       $OBUF = CURRENT POSITION IN OUTPUT BUFFER
                                ;       TEMP = TEMPLATE ADDRESS
                                ;
                                ;  OUTPUT:
                                ;       THE REMOTE NODE AREA IS WRITTEN INTO THE OUTPUT BUFFER
                                ;
                                ;-

000404                          SETARE:
000404  016700  000000G                 MOV     $OBUF,R0        ; GET CURRENT POSITION IN OUTPUT BUFFER
000410  016702  177364                  MOV     TEMP,R2         ; GET ADDRESS OF TEMPLATE
000414  016201  000010                  MOV     C.ADD(R2),R1    ; GET NODE ADDRESS
000420  042701  001777                  BIC     #1777,R1        ; ONLY WANT AREA
000424  000301                          SWAB    R1              ; SHIFT IT INTO LOW BYTE
000426  006201                          ASR     R1              ; NOT DONE YET
000430  006201                          ASR     R1              ; ...
000432  004767  000000G                 CALL    $CNVD           ; CONVERT NODE AREA TO DECIMAL
000436  010067  000000G                 MOV     R0,$OBUF        ; UPDATE POINTER INTO OUTPUT BUFFER
000442  000207                          RETURN

```

```
    160                                         .SBTTL  $UPROU - SETUP ROUTING DEFINITION
    161                                 ;+
    162                                 ; *** - $UPROU - SETUP ROUTING DEFINITION
    163                                 ;
    164                                 ; INPUT:
    165                                 ;       R3-R5   - TPARS REGISTERS
    166                                 ;
    167                                 ; OUTPUT:
    168                                 ;       C-BIT = SUCCESS/FAILURE
    169                                 ;
    170                                 ;-
    171
    172 000002                          $UPROU::
    173 000002  012700  000000G                 MOV     #$TMLST,R0      ; POINT AT START OF TEMPLATES
    174 000006  010001                  10$:    MOV     R0,R1           ; SAVE ADDRESS OF CURRENT TEMPLATE
    175 000010  011000                          MOV     (R0),R0         ; GET ADDRESS OF NEXT TEMPLATE
    176 000012  001441                          BEQ     101$            ; IF EQ, END OF TEMPLATES
    177 000014  122760  000000G 000000G         CMPB    #CS.ROU,C.STS(R0) ; IS THIS A ROUTING DEFINITION
    178 000022  001371                          BNE     10$             ; IF NE, NO .. KEEP LOOKING
    179                                 ;
    180                                 ; FOUND A TEMPLATE
    181                                 ;
    182 000024  011011                          MOV     (R0),(R1)       ; REMOVE TEMPLATE FROM CHAIN
    183 000026  010067  177746                  MOV     R0,TEMP         ; SAVE TEMPLATE ADDRESS
    184 000032  032760  000000G 000000G         BIT     #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
    185 000040  001004                          BNE     15$             ; IF NE, YES - UPDATE
    186 000042  012767  000000G 000000G         MOV     #$BUFI,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
    187 000050  000421                          BR      30$             ; AND LEAVE
    188 000052  062767  000004  177720  15$:    ADD     #C.ADD,TEMP     ; POINT AT NODE ADDRESS IN TEMPLATE
    189 000060  004767  000000G                 CALL    $MOVST          ; MOVE FIRST FIELD INTO OUTPUT BUFFER
    190 000064  010546                          MOV     R5,-(SP)        ; SAVE R5
    191 000066  005001                          CLR     R1              ; IGNORE BLANKS
    192 000070  012702  000000'                 MOV     #ROUKTB,R2      ; KEYWORD TABLE
    193 000074  012705  000000'                 MOV     #ROUSTB,R5      ; STATE TABLE
    194 000100  004767  000000G                 CALL    .TPARS          ; PARSE THE REST OF THE LINE
    195 000104  103002                          BCC     20$             ; NORMAL RETURN IF NO ERROR
    196 000106  005267  000000G                 INC     $ERROR          ; INDICATE SYNTAX ERROR
    197 000112  012605                  20$:    MOV     (SP)+,R5        ; RESTORE R5
    198 000114  000207                  30$:    RETURN
    199                                 ;
    200                                 ; ERROR CONDITIONS
    201                                 ;
    202 000116  000167  000000G         101$:   JMP     .CONER          ; CONSISTENCY ERROR
```

```
                                        .SBTTL  PUTOCT  - OUTPUT AN OCTAL NUMBER
                                ;+
                                ; ** PUTOCT - OUTPUT AN OCTAL NUMBER
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       JSR     R5,PUTOCT
                                ;       .WORD   OFFSET, CBTA OPTIONS, NUMBER OF BYTES, OPTION BIT
                                ;
                                ;-

000442                          PUTOCT: SAVRG   <R3>
000444  016700  000000G                 MOV     $OBUF,R0        ; POINT TO OUTPUT BUFFER
000450  012501                          MOV     (R5)+,R1        ; PICK UP OFFSET IN TEMPLATE
000452  066701  000000G                 ADD     .TEMP,R1        ; POINT TO VALUE TO CONVERT
000456  012502                          MOV     (R5)+,R2        ; PICK UP OPTIONS
000460  012503                          MOV     (R5)+,R3        ; COUNT
000462  032577  000000G                 BIT     (R5)+,@.FLAGS   ; OPTION SELECTED ?
000466  001402                          BEQ     10$             ; IF EQ, NOPE
000470  004767  000000G                 CALL    $CBTA3          ; CONVERT BINARY TO ASCII

000474  112720  000054          10$:    MOVB    #<',>,(R0)+     ; STUFF A COMMA
000500  010067  000000G                 MOV     R0,$OBUF        ; RESTORE BUFFER POINTER
000504                                  RESRG   <R3>
000506  000205                          RTS     R5              ; FINIS

```

```
                                                .SBTTL  SETNAM - CHECK DDM PROCESS NAME
                                        ;+
                                        ; *** - SETNAM - CHECK DDM PROCESS NAME
                                        ;
                                        ; INPUT:
                                        ;       .PSTPT = START ADDRESS OF THE ASCII PROCESS NAME
                                        ;       .PSTCN = NUMBER OF CHARS IN THE PROCESS NAME
                                        ;       (.TEMP) = TEMPLATE ADDRESS
                                        ;       (.TEMP)+C.NAM = DDM PROCESS ASSOCIATED WITH THIS SLT
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-
000204                                  SETNAM:
000204  004767  000000G                         CALL    $MOVPT          ; MOVE THE PROCESS NAME INTO OUTPUT BUFFER
000210  016700  000000G                         MOV     .PSTPT,R0       ; GET ADDRESS OF ASCII PROCESS NAME
000214  010701                                  MOV     PC,R1           ; PERIODS ARE ACCEPTABLE
000216  004767  000000G                         CALL    $CAT5           ; CONVERT PROCESS NAME TO RAD50
000222  016700  000000G                         MOV     .TEMP,R0        ; GET START OF TEMPLATE
000226  020160  000004                          CMP     R1,C.NAM(R0)    ; IS THIS THE RIGHT TEMPLATE ?
000232  001402                                  BEQ     10$             ; IF EQ, YES .. OKAY
000234  062716  000002                          ADD     #2,(SP)         ; ELSE, REJECT THE TRANSITION
000240  000207                          10$:    RETURN
```

```
        .TITLE  UPDSTA - CFE UPDATE STATION DEFINITION IN CETAB
        .IDENT  /V05.00/
        .ENABL  LC
;
;
; COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       STATE TABLE TO PARSE THE STATION DEFINITIONS
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;


```

```
**FILE**ID**UPDSVC










UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   VV      VV       CCCCCCCC
UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   VV      VV       CCCCCCCC
UU      UU   PP      PP   DD      DD   SS           VV      VV     CC
UU      UU   PP      PP   DD      DD   SS           VV      VV     CC
UU      UU   PP      PP   DD      DD   SS           VV      VV     CC
UU      UU   PP      PP   DD      DD   SS           VV      VV     CC
UU      UU   PPPPPPPP     DD      DD     SSSSSS     VV      VV     CC
UU      UU   PPPPPPPP     DD      DD     SSSSSS     VV      VV     CC
UU      UU   PP           DD      DD         SS   VV      VV     CC
UU      UU   PP           DD      DD         SS   VV      VV     CC
UU      UU   PP           DD      DD         SS     VV  VV       CC               ....
UU      UU   PP           DD      DD         SS     VV  VV       CC               ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS         VV         CCCCCCCC       ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS         VV         CCCCCCCC       ....





LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
UPDSVC - UPDATE SWITCHED VIRTUA MACRO V05.03b Saturday 29-Jun-85 05:58
Table of contents
```

```
 55                         ;****
 56                         ; LOCAL MACROS
 57                         ;****
 58
 59                                 .MACRO  PRINT   TEXT
 60                                 MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
 61                                 CALL    $PRINT          ; PRINT MESSAGE
 62                                 .ENDM   PRINT
 63
 64                         ;
 65                         ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
 66                         ;
 67                                 .MACRO  $$$PTN  ROU
 68                                 .MACRO  'ROU    OFS,OCTL,OPT,?L
 69                                 .IF NB <OPT>
 70                                 MOV     OPT,R2
 71                                 CALL    $PTOPT
 72                                 BCS     L
 73                                 .IFTF
 74                                 MOV     OFS,R2
 75                                 .IF NB <OCTL>
 76                                 MOV     SP,R1
 77                                 .IFF
 78                                 CLR     R1
 79                                 .ENDC ; IF NB <OCTL>
 80                                 CALL    'ROU
 81                                 .IFT  ; IF NB <OPT>
 82                         L:
 83                                 .ENDC ; IF NB <OPT>
 84                                 .ENDM ; 'ROU
 85                                 .ENDM ; $$$PTN
 86
 87 000000                          $$$PTN  $PTNUM
 88 000000                          $$$PTN  $PTNMB
 89
 90                         ;****
 91                         ; MACRO CALLS
 92                         ;****
 93
 94                                 .MCALL  ISTAT$,STATE$,TRAN$
 95
 96                         ;****
 97                         ; LOCAL DATA
 98                         ;****
 99
100                         ;
101                         ; DEFINE UNT TEMPLATE
102                         ;
103 000000                          .ASECT
104         000000                  .=0
105 000000                          .BLKW   1               ; C.LNK
106 000002                          .BLKW   1               ; C.STS
107 000004                  C.UNT:  .BLKW   1               ; UNIT NUMBER
108 000006                  C.CW0:  .BLKW   1               ; CHARACTERISTICS WORD 0
109 000010                  C.CW1:  .BLKW   1               ; CHARACTERISTICS WORD 1
110 000012                  C.CSR:  .BLKW   1               ; SECONDARY CSR ADDRESS
111 000014                  C.CST:  .BLKW   1               ; LINE COST
```

```
UPDUNT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:58        PAGE 2
SYMBOL CROSS REFERENCE                                      CREF   04.00
SYMBOL  VALUE              REFERENCES

.FLAGS  = ******  GX       *6-178
.PNUMB  = ******  GX        7-213
.TEMP   = ******  GX       *6-176       7-212       8-232       9-254
.TPARS  = ******  GX        6-185
```

Table of contents

```
     4-   57    MACRO CALLS AND LOCAL DATA
     5-  122    LOCAL DATA
     6-  130    $UPX29 - UPDATE X.29 CHARACTERISTICS
```

```
                                          ;
                                                  .MACRO  $PTR50  OFS,WC
                                                  MOV     OFS,R2
                                                  MOV     WC,R1
                                                  CALL    $PTR50
                                                  .ENDM

                                          ;
                                          ; DEFINE THE X.25 LEVEL 3 PARAMETER BLOCK
                                          ;
000000                                            .ASECT
        000000                                    .=0
000000                                            .BLKW   1               ; C.LNK
000002                                            .BLKW   1               ; C.STS
000004                                    C.DFP:  .BLKW   1               ; DEFAULT BLOCK SIZE    (21.-1029.)
000006                                    C.MXP:  .BLKW   1               ; MAXIMUM BLOCK SIZE    (21.-1029.)
000010                                    C.DW:   .BLKB   1               ; DEFAULT WINDOW SIZE   (1.-127.)
000011                                    C.MW:   .BLKB   1               ; MAXIMUM WINDOW SIZE   (1.-127.)
000012                                    C.CT:   .BLKB   1               ; CALL TIMER VALUE      (0.-255.)
000013                                    C.KT:   .BLKB   1               ; CLEAR TIMER VALUE     (0.-255.)
000014                                    C.RT:   .BLKB   1               ; RESET TIMER VALUE     (0.-255.)
000015                                    C.ST:   .BLKB   1               ; RESTART TIMER VALUE   (0.-255.)
000016                                    C.KM:   .BLKB   1               ; MAXIMUM CLEAR VALUE   (0.-255.)
000017                                    C.RM:   .BLKB   1               ; MAXIMUM RESETS VALUE  (0.-255.)
000020                                    C.SM:   .BLKB   1               ; MAXIMUM RESTARTS VALUE(0.-255.)
        000021                            C.LEN=.                         ; LENGTH OF THE BLOCK
000000                                            .PSECT
                                                  .NLIST  BEX
000000     011      130     063           X3PDF:  .ASCII  <11>/X3P$DF/<11>
        000010                                    X3PLN=.-X3PDF
                                                  .LIST   BEX
                                                  .EVEN

```

```
    218
    219                                          .IFT
    220
    221 002006                                   FSRSZ$  0,,MLDCOD             ; Define FCS work area
    222
    223                                          .IFTF
    224
    225                                 ;        Executive Vector Table
    226
    227 000000                          EXEVEC:
    228 000000   000000                          .WORD   0
    229 000002   000000G                ISAR5::  .WORD   KISAR5
    230 000004   000000G                BLXIO::  .WORD   $BLXIO
    231                                          .IF     DF      R$$MPL       ; If system is RSX-11M+ ...
    232                                 NCPU::   .WORD   $NCPU
    233                                          .ENDC
    234 000006   000000G                NTUIC::  .WORD   $NTUIC
    235 000010   000000G                PARHD::  .WORD   $PARHD
    236 000012   000000G                QRMVF::  .WORD   $QRMVF
    237 000014   000000G                TKTCB::  .WORD   $TKTCB
    238 000016   000000G                CCBRT::  .WORD   $CCBRT
    239 000020   000000G                LDBGT::  .WORD   $LDBGT
    240 000022   000000G                LDBRT::  .WORD   $LDBRT
    241 000024   000000G                PDSPL::  .WORD   $PDSPL
    242 000026   000000G                PDVTA::  .WORD   $PDVTA
    243 000030   000000G                RDBSZ::  .WORD   $RDBSZ
    244 000032   000000G                SLTMA::  .WORD   $SLTMA
    245 000034   000000G                STDDM::  .WORD   $STDDM
    246
    247          000017                 EXEVCL  = .-EXEVEC/2
```

```
                .SBTTL  LOCPAR  - Locate Microcode Partition
        ;+
        ;                               *** LOCPAR ***
        ;
        ; Function:     To locate a (sub)partition that contains the microcode
        ;               specified in the current load request.  If the partition
        ;               is found,  it's PCB address is saved for later use.
        ;
        ; Call Format:  CALL    LOCPAR
        ;
        ; Inputs:       R4 = address of microcode partition ASCII name
        ;               R5 = length of ASCII string
        ;
        ; Outputs:      C-bit clear = partition found
        ;               C-bit set   = partition not found
        ;
        ;               If the partition has been located, then:
        ;
        ;               BLOCK   = mapping bias of partition
        ;               BLOCK+2 = length of partition in 32-word blocks
        ;
        ; Registers:    R0 through R3 destroyed
        ;
        ;-

        LOCPAR: MOV     R4,R3                           ; Copy address of string
                ADD     R5,R3                           ; Point past end of string
                MOVB    (R3),-(SP)                      ; Save whatever is there
                MOVB    #11,(R3)                        ; Put a known delimiter in its place
                MOV     R3,-(SP)                        ; Save the delimiter address
                CLR     -(SP)                           ; Allocate two words on the stack
                CLR     -(SP)                           ;   to store RAD50 partition name

                MOV     #1,R1                           ; Allow periods to be in partition name
                MOV     R4,R0                           ; Copy address of ASCII string
                CALL    $CAT5                           ; Convert three characters to RAD50
                MOV     R1,(SP)                         ; Move RAD50 equivalent to save area
                BCS     10$                             ; Branch if less than three converted
                MOV     #1,R1                           ; Allow periods to be in partition name
                CALL    $CAT5                           ; Convert three more characters
                MOV     R1,2(SP)                        ; Move RAD50 equivalent to save area

        10$:    MOV     SP,R3                           ; Copy address of RAD50 partition name
                CLR     PCBADR                          ; Clear local PCB address

                .IF     DF      P$$LAS                  ; If system has PLAS support ...

                CALL    $SRNAM                          ; Use executive routine to find PCB

                .IFF                                    ; If no PLAS support ...

                CALL    SRNAM                           ; Use internal routine to find PCB

                .IFTF                                   ; Regardless of PLAS support ...

                BCC     20$                             ; Branch if partition found

```

```
                                                                              I 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 5
SYMBOL  CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE               REFERENCES

LF.MFL   = 004000             #6-85
LF.MTP   = 000020             #6-85
LF.PAC   = 000200             #6-85
LF.RDY   = 040000             #6-85
LF.REA   = 010000             #6-85
LF.SER   = 000040             #6-85
LF.TIM   = 000010             #6-85
LF.UNL   = 020000             #6-85
LF.X2P   = 000000             #6-85
LN.CLO   = 000000             #6-85
LN.DUM   = 000005             #6-85
LN.LOA   = 000004             #6-85
LN.LOO   = 000003             #6-85
LN.OAU   = 000003             #6-85
LN.OFF   = 000001             #6-85
LN.ON    = 000000             #6-85
LN.OOP   = 000004             #6-85
LN.OPE   = 000001             #6-85
LN.REF   = 000002             #6-85
LN.SER   = 000002             #6-85
LN.STA   = 000017             #6-85
LN.SUB   = 000360             #6-85
LN.TRI   = 000006             #6-85
LOCFIL     001372 R            9-434     #14-752
LOGEVT     001022 R           10-574     #11-599      13-724
L.COST     000015             #6-85
L.CTL      000012             #6-85       9-405
L.CVA      177776             #6-85
L.DDM      000002             #6-85       9-403
L.DDS      000004             #6-85       9-404
L.DLC      000003             #6-85
L.DLM      000006             #6-85
L.DLS      000010             #6-85
L.FLG      000000             #6-85
L.KRBA     000016             #6-85
L.LEN    = 000022             #6-85
L.MPF      000022             #6-85
L.NMST     000020             #6-85
L.NSTA     000014             #6-85
L.OWNR     000021             #6-85
L.UNT      000013             #6-85
MLD        000036 R           #9-269      18-989
MLDERR     000000 R           #7-147      11-607
MLOAD      001506 R            9-451     #15-793      15-806
MO$SAC   = 000016             #6-83
MO$SPR   = 000012             #6-83
MO$SSV   = 000014             #6-83
MO$25A   = 000006             #6-83
MO$25P   = 000002             #6-83
MO$25S   = 000004             #6-83
MO$29S   = 000010             #6-83
MXFNLN   = 000006             #7-108      14-752      14-754
```

```
                                                                              J 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 6
```

```
                                        .SBTTL  Load Control Data
                                        .IFT

                                ;       Input File Descriptor Block (FDB) Definitions

                                FDB:    FDBDF$                                    ; Allocate an FDB block
                                        FDAT$A                                    ; File already exists
                                        FDRC$A  FD.RWM                            ; Read/write block I/O access
                                        FDBK$A  FCSBUF,FCSBSZ,,FCSEFL,FCSIOS      ; Set buffer/size, flag, status
                                        FDOP$A  FCSLUN,DSDS                       ; Specify LUN and filespec area

                                ;       General Data Declarations

                                FCSBUF: .BLKB   FCSBSZ                  ; Output file I/O buffer
                                FCSIOS: .BLKW   2                       ; FCS I/O status doubleword
                                FILERR: .BLKW   1                       ; FCS error storage for $EDMSG
                                DSDS:   .WORD   0                       ; Length of source device name
                                        .WORD   FILDEV                  ; Source device name address
                                        .WORD   0                       ; Length of source directory
                                        .WORD   FILDIR                  ; Source directory address
                                        .WORD   0                       ; Length of source filename
                                        .WORD   0                       ; Address of source filename
                                FILDEV: .BLKB   6                       ; Source device
                                FILDIR: .BLKB   10.                     ; Source directory

                                        .IF     DF R$$MPL
                                        .IF     NDF R$$PRO

                                GINVEC: GIN$    GI.VEC,EXEVEC,EXEVCL

                                        .ENDC   ; NDF R$$PRO
                                        .ENDC   ; DF R$$MPL

                                        .IFF

000124                          PCBADR: .BLKW   1                       ; Microcode PCB address

                                        .IFTF

                                        .IF     DF      N$$UMR

000126                          SETTMR: MRKT$   TMREFL,LDBWTS,2         ; DPB for LDB wait timer
000140                          WAIT:   STSE$   TMREFL                  ; DPB to stop task for TMREFL
000144                          LDBPTR: .BLKW   1                       ; Address of CCB/LDB
000146                          LDBIAS: .BLKW   1                       ; Mapping bias of LDB
000150                          LDBADR: .BLKW   1                       ; Starting address of LDB

                                        .ENDC

000152  000000  000000          BLOCK:  .WORD   0,0                     ; Block number for load control
000156                          CURQST: .BLKW   1                       ; Address of current load CCB
000160                          PDVIDX: .BLKW   1                       ; PDV index of device driver
000162                          CTLNUM: .BLKW   1                       ; Device controller number
000164                          DDMTBL: .BLKW   1                       ; Driver's line table address
000166                          WORKSP: .BLKB   80.                     ; Work space
000306  000000  000000  000000  EVTBLK: .WORD   0,0,0,2,0,0,0,0         ; Event descriptor block
000326                                  .BLKB   160.                    ; Event data area
```

```
; PUT RAD50 TEMPLATE FIELD TO BUFFER
;
        .MACRO  $PTR50  OFS,WC,OPT,?L
        .IF NB <OPT>
        MOV     OPT,R2
        CALL    $PTOPT
        BCS     L
        .IFTF
        MOV     OFS,R2
        MOV     WC,R1
        CALL    $PTR50
        .IFT
L:
        .ENDC
        .ENDM

```

```
                                                  .SBTTL  $UPRDT - UPDATE DESTINATION DESCRIPTOR
                                          ;+
                                          ; *** - $UPRDT - UPDATE DESTINATION DESCRIPTOR
                                          ;
                                          ; INPUT:
                                          ;       R3-R5   - TPARS REGISTERS
                                          ;
                                          ; OUTPUT:
                                          ;       C-BIT = SUCCESS/FAILURE
                                          ;
                                          ;-
                                          
       000010                             $UPRDT::
       000010  012767  000000G 000000C            MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
       000016  010567  000000G                    MOV     R5,.TEMP         ; SAVE TEMPLATE ADDRESS
       000022  005725                             TST     (R5)+            ; POINT TO C.STS
       000024  010567  000000G                    MOV     R5,.FLAGS        ; SAVE ADDRESS OF FLAGS WORD
       000030  012767  177777  000000G            MOV     #-1,.COMMA       ; INITIALIZE FOR OPTIONS
       000036  012700  000000G                    MOV     #.BUFF,R0        ; POINT AT START OF OUTPUT BUFFER
       000042  012701  000000'                    MOV     #RDTDF,R1        ; POINT AT START OF RDT DEFINITION
       000046  012702  000010                     MOV     #RDTLN,R2        ; GET LENGTH OF RDT DEFINITION
       000052  004767  000000G                    CALL    $MVASC           ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
       000056  010067  000000G                    MOV     R0,$OBUF         ; SAVE CURRENT POSITION IN OUTPUT BUFFER
       000062                                     $PTR50  #C.NAM,#2        ; SET NAME
       000076                                     $PTR50  #C.ADR,#5        ; SET ADDRESS
       000112  016701  000000G                    MOV     $OBUF,R1         ; CALCULATE LENGTH OF BUFFER
       000116  162701  000000G                    SUB     #.BUFF,R1        ;       ...
       000122  010167  000000C                    MOV     R1,$OFDB+F.NRBD  ; SET LENGTH OF BUFFER
       000126  004767  000000G                    CALL    $WDATA           ; WRITE THE RECORD TO THE OUTPUT FILE
       000132  012767  000000G 000000C            MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE ''END$DF'' LINE
       000140  000207                             RETURN
                                          
                                          
               000001                             .END
```

```
                                   .SBTTL  SETADD - WRITE OUT REMOTE NODE ADDRESS
                           ;+
                           ; *** - SETADD - WRITE OUT REMOTE NODE ADDRESS
                           ;
                           ; INPUT:
                           ;       $OBUF = CURRENT POSITION IN OUTPUT BUFFER
                           ;       TEMP = TEMPLATE ADDRESS
                           ;
                           ; OUTPUT:
                           ;       THE REMOTE NODE ADDRESS IS WRITTEN INTO THE OUTPUT BUFFER
                           ;
                           ;-

000444                     SETADD:
000444  016700  000000G            MOV     $OBUF,R0        ; GET CURRENT POSITION IN OUTPUT BUFFER
000450  016702  177324             MOV     TEMP,R2         ; GET ADDRESS OF TEMPLATE
000454  016201  000010             MOV     C.ADD(R2),R1    ; GET NODE ADDRESS
000460  042701  176000             BIC     #^C1777,R1      ; CLEAR AREA
000464  004767  000000G            CALL    $CNVD           ; CONVERT NODE ADDRESS TO DECIMAL
000470  010067  000000G            MOV     R0,$OBUF        ; UPDATE POINTER INTO OUTPUT BUFFER
000474  000207                     RETURN

```

```
                                        .SBTTL  SETNUM - SET THE NEXT NUMBER
                                ;+
                                ; *** - SETNUM - SET THE NEXT NUMBER
                                ;
                                ; INPUT:
                                        $OBUF = CURRENT POSITION IN OUTPUT LINE
                                        TEMP = TEMPLATE ADDRESS
                                ;
                                ; OUTPUT:
                                        C-BIT = SUCCESS/FAILURE
                                        $OBUF = NEXT LOCATION IN OUTPUT BUFFER
                                        TEMP = ADDRESS OF NEXT LOCATION IN THE TEMPLATE
                                ;
                                ;-

000122                          SETNUM:
000122  016700  000000G                 MOV     $OBUF,R0                ; POINT AT CURRENT POSITION IN LINE
000126  016702  177646                  MOV     TEMP,R2                 ; POINT AT TEMPLATE
000132  012201                          MOV     (R2)+,R1                ; GET THE NUMBER
000134  010267  177640                  MOV     R2,TEMP                 ; SAVE NEXT LOCATION IN TEMPLATE
000140  004767  000000G                 CALL    $CNVD                   ; CONVERT THE NUMBER TO DECIMAL
000144  010067  000000G                 MOV     R0,$OBUF                ; UPDATE POINTER INTO OUTPUT LINE
000150  000207                          RETURN


        000001                          .END
```

```
                                         .SBTTL  PUTFLG - OUTPUT FLAGS
                                ;+
                                ; ** PUTFLG - OUTPUT FLAGS
                                ;
                                ; INPUT:         CALL VIA:       JSR R5,PUTFLG
                                ;                                .WORD   TEXTADDR_1,FLAG_1
                                ;                                  :
                                ;                                .WORD   TEXTADDR_N,FLAG_N
                                ;                                .WORD   0
                                ;
                                ;
                                ;
                                ; OUTPUT:        FLAGS OUTPUT, R0 UPDATED.
                                ;-

000510  016700  000000G         PUTFLG: MOV     $OBUF,R0        ; GET BUFFER POINTER
000514  012501                  10$:    MOV     (R5)+,R1        ; GET TEXT ADDRESS
000516  001407                          BEQ     30$             ; IF EQ, DONE
000520  032577  000000G                 BIT     (R5)+,@.FLAGS   ; FLAG PRESENT ?
000524  001773                          BEQ     10$             ; NO - KEEP LOOKING
000526  112120                  20$:    MOVB    (R1)+,(R0)+     ; ELSE MOVE TEXT
000530  001376                          BNE     20$             ; ...
000532  005300                          DEC     R0              ; BACK OVER NUL BYTE
000534  000767                          BR      10$             ; KEEP LOOKING

000536  124027  000041          30$:    CMPB    -(R0),#<'!>     ; LAST CHARACTER A "!"
000542  001401                          BEQ     40$             ; YES
000544  005200                          INC     R0              ; NO
000546  112720  000054          40$:    MOVB    #<',>,(R0)+     ; INSERT DELIMITER
000552  010067  000000G                 MOV     R0,$OBUF        ; RESTORE BUFFER POINTER
000556  000205                          RTS     R5

        000001                          .END
```

```
                                                .SBTTL  SETFLG - SETUP THE FLAGS WORD
                                        ;+
                                        ; *** - SETFLG - SETUP THE FLAGS WORD
                                        ;
                                        ; INPUT:
                                        ;       (.TEMP) = TEMPLATE ADDRESS
                                        ;       (.TEMP)+C.FLG = FLAGS WORD
                                        ;
                                        ; OUTPUT:
                                        ;
                                        ;-

000242                                  SETFLG:
000242  010446                                  MOV     R4,-(SP)            ; SAVE R4
000244  016704  000000G                         MOV     .TEMP,R4            ; GET START OF TEMPLATE
000250  016700  000000G                         MOV     $OBUF,R0            ; POINT R0 AT CURRENT POSITION IN OUTPUT BUFFER
000254  032764  000000G 000006                  BIT     #LF.MDC,C.FLG(R4)   ; DOES THE LINE NEED MODEM CONTROL ?
000262  001406                                  BEQ     20$                 ; IF EQ, NO .. CONTINUE
000264  012701  000007'                         MOV     #LMDC,R1            ; POINT AT INPUT STRING
000270  012702  000007                          MOV     #LMDCL,R2           ; AND LENGTH OF STRING
000274  004767  000000G                         CALL    $MVASC              ; SET MDC BIT IN SLT
000300  032764  000000G 000006  20$:            BIT     #LF.MTP,C.FLG(R4)   ; IS THE LINE MULTI-POINT ?
000306  001406                                  BEQ     30$                 ; IF EQ, NO .. CONT.
000310  012701  000016'                         MOV     #LMTP,R1            ; POINT AT INPUT STRING
000314  012702  000007                          MOV     #LMTPL,R2           ; AND LENGTH OF STRING
000320  004767  000000G                         CALL    $MVASC              ; SET MTP BIT IN SLT
000324  032764  000000G 000006  30$:            BIT     #LF.MFL,C.FLG(R4)   ; IS THE LINE MARKED FOR LOAD ?
000332  001406                                  BEQ     35$                 ; IF EQ, NO .. CONT.
000334  012701  000025'                         MOV     #LMFL,R1            ; POINT AT INPUT STRING
000340  012702  000007                          MOV     #LMFLL,R2           ; AND LENGTH OF STRING
000344  004767  000000G                         CALL    $MVASC              ; SET MFL BIT IN SLT
000350  032764  000000G 000006  35$:            BIT     #LF.ENA,C.FLG(R4)   ; IS THE LINE TO BE ENABLED ?
000356  001406                                  BEQ     40$                 ; IF EQ, NO
000360  012701  000000'                         MOV     #LENA,R1            ; POINT AT INPUT STRING
000364  012702  000007                          MOV     #LENAL,R2           ; AND LENGTH
000370  004767  000000G                         CALL    $MVASC              ; SET ENA BIT IN SLT
000374  032764  000000G 000006  40$:            BIT     #LF.TIM,C.FLG(R4)   ; DOES THE LINE NEED TIMER SERVICE ?
000402  001406                                  BEQ     50$                 ; IF EQ, NO .. CONT.
000404  012701  000034'                         MOV     #LTIM,R1            ; POINT AT INPUT STRING
000410  012702  000007                          MOV     #LTIML,R2           ; AND LENGTH OF STRING
000414  004767  000000G                         CALL    $MVASC              ; SET TIM BIT IN SLT
000420  032764  000000G 000006  50$:            BIT     #LF.X2P,C.FLG(R4)   ; DOES THE LINE NEED X.25 LEVEL 2 ?
000426  001406                                  BEQ     60$                 ; IF EQ, NO .. CONT.
000430  012701  000043'                         MOV     #LX2P,R1            ; POINT AT INPUT STRING
000434  012702  000007                          MOV     #LX2PL,R2           ; AND LENGTH OF STRING
000440  004767  000000G                         CALL    $MVASC              ; SET TIM BIT IN SLT
000444  032764  000000G 000006  60$:            BIT     #LF.BRO,C.FLG(R4)   ; ETHERNET ?
000452  001406                                  BEQ     65$                 ; IF EQ, NO .. CONT.
000454  012701  000052'                         MOV     #LBRO,R1            ; POINT AT INPUT STRING
000460  012702  000007                          MOV     #LBROL,R2           ; AND LENGTH OF STRING
000464  004767  000000G                         CALL    $MVASC              ; SET TIM BIT IN SLT
000470  032764  000000G 000006  65$:            BIT     #LF.SER,C.FLG(R4)   ; SERVICE DISABLED ?
000476  001406                                  BEQ     70$                 ; IF EQ, NO .. CONT.
000500  012701  000061'                         MOV     #LSER,R1            ; POINT AT INPUT STRING
000504  012702  000007                          MOV     #LSERL,R2           ; AND LENGTH OF STRING
000510  004767  000000G                         CALL    $MVASC              ; SET TIM BIT IN SLT
000514  020067  000000G         70$:            CMP     R0,$OBUF            ; ANY FLAGS ADDED ?
```

```
     55                                                  ;****
     56                                                  ; LOCAL MACROS
     57                                                  ;****
     58
     59                                                          .MACRO  PRINT   TEXT
     60                                                          MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
     61                                                          CALL    $PRINT          ; PRINT MESSAGE
     62                                                          .ENDM   PRINT
     63
     64                                                  ;****
     65                                                  ; MACRO CALLS
     66                                                  ;****
     67
     68                                                          .MCALL  ISTAT$,STATE$,TRAN$
     69
     70
     71                                                  ;****
     72                                                  ; LOCAL DATA
     73                                                  ;****
     74
     75 000000                                           TEMP:   .BLKW   1               ; ADDRESS OF STATION TEMPLATE
     76 000002        123       106       056            SENA:   .ASCII  /SF.ENA!/       ; STATION IS TO BE ENABLED
        000005        105       116       101
        000010        041
     77            000007                                        SENAL=.-SENA
     78 000011        123       106       056            SMFL:   .ASCII  /SF.MFL!/       ; STATION IS MARKED FOR LOAD
        000014        115       106       114
        000017        041
     79            000007                                        SMFLL   = .-SMFL
     80 000020        123       106       056            SSVC:   .ASCII  /SF.SVC!/       ; STATION IS AN SVC
        000023        123       126       103
        000026        041
     81            000007                                        SSVCL   = .-SSVC
     82 000027        123       106       056            SSER:   .ASCII  /SF.SER!/       ; SERVICE REQUESTS DISABLED ON CIRCUIT
        000032        123       105       122
        000035        041
     83            000007                                        SSERL   = .-SSER
     84                                                          .EVEN
     85                                                  ;
     86                                                  ; DEFINE STATION TEMPLATE
     87                                                  ;
     88 000000                                                   .ASECT
     89            000000                                        .=0
     90 000000                                                   .BLKW   1               ; C.LNK
     91 000002                                                   .BLKW   1               ; C.STS
     92 000004                                           C.STA:  .BLKB   1               ; STATION NUMBER
     93 000005                                           C.ADD:  .BLKB   1               ; STATION ADDRESS
     94 000006                                           C.FLG:  .BLKB   1               ; STATION FLAGS
     95 000007                                           C.CST:  .BLKB   1               ; STATION COST
     96 000010                                           C.APR:  .BLKB   1               ; ACTIVE POLLING RATIO
     97 000011                                                   .BLKB   1               ; AVAILABLE
     98 000012                                           C.HTM:  .BLKW   1               ; HELLO TIMER
     99 000014                                                   .BLKW   1               ; UNUSED
    100            000016                                C.LEN=.
    101 000036                                                   .PSECT
```

Table of contents

```
     4-   57    MACRO CALLS AND LOCAL DATA
     4-   58    MACRO DEFINITIONS
     5-  131    LOCAL DATA
     6-  144    $UPSVC - UPDATE SWITCHED VIRTUAL CIRCUIT
```

```
    112 000016                          C.DPR:  .BLKW   1               ; DEAD POLLING RATIO
    113 000020                                  .BLKW   1               ; UNUSED
    114 000022                                  .BLKW   1               ; UNUSED
    115 000024                          C.PCH:  .BLKW   1               ; EMULATOR CHARACTERISTICS
    116         000026                  C.LEN=.
    117 000000                                  .PSECT
    118
```

```
UPDUNT      CREATED BY  MACRO  ON 29-JUN-85 AT 05:58     PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

DBGTP$           #5-124
ISTAT$           #4-94        5-124
MTRAN$           #5-124
PRINT            #4-59
STATE$           #4-94        5-126      #5-129     #5-132     #5-135     #5-138     #5-142     #5-144     #5-146     #5-148
                 #5-151
TRAN$            #4-94       #5-127      #5-130     #5-133     #5-136     #5-139     #5-143     #5-145     #5-147     #5-149
$PTNMB           #4-88
$PTNUM           #4-87        10-272     11-286     12-300     13-314
$$$PTN           #4-67        4-87       4-88
```

```
**FILE**ID**UPDX2P
```

```
        .TITLE  UPDX29 - UPDATE X.29 3 CHARACTERISTICS
        .IDENT  /V05.00/
        .ENABL  LC

;
; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       UPDATE X.29 CHARACTERISTICS IN CETAB
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-AUG-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                                .SBTTL  $UPX3P - UPDATE X.25 LEVEL 3 CHARACTERISTICS
                                        ;+
                                        ; *** - $UPX3P - UPDATE X.25 LEVEL 3 CHARACTERISTICS
                                        ;
                                        ; INPUT:
                                        ;       R5 - ADDRESS OF TEMPLATE
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000010                                  $UPX3P::
000010  012767  000000G 000000C                 MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
000016  010567  000000G                         MOV     R5,.TEMP        ; SAVE UPDATED TEMPLATE POINTER
000022  005725                                  TST     (R5)+           ; POINT TO C.STS
000024  010567  000000G                         MOV     R5,.FLAGS       ; SAVE ADDRESS OF FLAGS WORD
000030  012767  177777  000000G                 MOV     #-1,.COMMA      ; SUPPRESS INITIAL COMMA
000036  012700  000000G                         MOV     #.BUFF,R0       ; POINT AT START OF OUTPUT BUFFER
000042  012701  000000'                         MOV     #X3PDF,R1       ; POINT AT START OF X3P DEFINITION
000046  012702  000010                          MOV     #X3PLN,R2       ; GET LENGTH OF X3P DEFINITION
000052  004767  000000G                         CALL    $MVASC          ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
000056  010067  000000G                         MOV     R0,$OBUF        ; SAVE CURRENT POSITION IN OUTPUT BUFFER
000062                                          $PTNUM  #C.DFP          ; DEFAULT BLOCK SIZE
000074                                          $PTNUM  #C.MXP          ; MAX BLOCK SIZE
000106                                          $PTNMB  #C.DW           ; DEFAULT WINDOW SIZE
000120                                          $PTNMB  #C.MW           ; MAX WINDOW SIZE
000132                                          $PTNMB  #C.CT           ; CALL TIMER
000144                                          $PTNMB  #C.KT           ; CLEAR TIMER
000156                                          $PTNMB  #C.RT           ; RESET TIMER
000170                                          $PTNMB  #C.ST           ; RESTART TIMER
000202                                          $PTNMB  #C.KM           ; MAX CLEAR VALUE
000214                                          $PTNMB  #C.RM           ; MAX RESETS
000226                                          $PTNMB  #C.SM           ; MAX RESTARTS
000240  016701  000000G                         MOV     $OBUF,R1        ; CALCULATE LENGTH OF BUFFER
000244  162701  000000G                         SUB     #.BUFF,R1       ;       ...
000250  010167  000000C                         MOV     R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
000254  004767  000000G                         CALL    $WDATA          ; WRITE THE RECORD TO THE OUTPUT FILE
000260  012767  000000G 000000C                 MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE ''END$DF'' LINE
000266  000207                                  RETURN


        000001                                  .END
```

```
                                            .SBTTL  MLD     - Microcode Loader Main Routine

000036                                      .PSECT  MLDCOD,I,RO,REL,OVR,LCL
                                    ;+
                                                                    *** MLD ***
                                    ;
                                    ; Function:     This program is used service requests from DECnet device
                                    ;               drivers for specific microcode files (RSX-11M) or partitions
                                    ;               (RSX-11S).
                                    ;
                                    ; Inputs:       No explicit inputs.  The task's receive listhead in the TCB
                                    ;               should be non-empty when the task begins execution.  Load
                                    ;               requests are queued to this list by AUX upon request from
                                    ;               device drivers.
                                    ;
                                    ; Outputs:      No explicit outputs.  MLD will perform a single load session
                                    ;               with each driver that has made a request.  Requests are
                                    ;               handled FIFO with no multiplexing.
                                    ;
                                    ;-

000036                              MLD:    ALUN$S  #EVTLUN,#"NM,#0         ; Assign LUN to network management

000056                              1$:     .IFT                            ; If system is RSX-11M ...

                                            .IF     DF R$$MPL               ; If system is RSX-11M-PLUS
                                            .IF     NDF R$$PRO              ; And not P/OS

                                            DIR$    #GINVEC                 ; Initialize executive vector table
                                            BCC     2$                      ; Branch if no error
                                            CALL    ABOAST                  ; Else , clean up and exit
                                    2$:
                                            .ENDC   ; NDF R$$PRO
                                            .ENDC   ; DF R$$MPL

000056                                      FINIT$                          ; Initialize FCS impure area
000062  017700  177726                      MOV     @TKTCB,R0               ; Get our TCB address
000066  016000  000044                      MOV     T.LDV(R0),R0            ; Get our load device UCB address
000072  016003  000000                      MOV     U.DCB(R0),R3            ; Get load device DCB address
000076  016367  000004  001326'             MOV     D.NAM(R3),FILDEV        ; Copy load device name
000104  166300  000002                      SUB     D.UCB(R3),R0            ; Compute relative UCB address
000110  016301  000010                      MOV     D.UCBL(R3),R1           ; Get UCB length
000114                                      CALL    $DIV                    ; Compute relative unit number
000120  066300  000006                      ADD     D.UNIT(R3),R0           ; Compute absolute unit number
000124  042700  177400                      BIC     #177400,R0              ; Clear high byte
000130  010001                              MOV     R0,R1                   ; Copy it
000132  012700  001330'                     MOV     #FILDEV+2,R0            ; Get address of ASCII unit number
000136  012702  014010                      MOV     #CVTPRM,R2              ; Get conversion parameters
000142                                      CALL    $CBTA                   ; Convert unit to ASCII
000146  112720  000072                      MOVB    #':,(R0)+               ; Complete the device name
000152  162700  001326'                     SUB     #FILDEV,R0              ; Get length of device name
000156  010067  001312'                     MOV     R0,DSDS                 ; Move it to dataset descriptor

000162  012702  001334'                     MOV     #FILDIR,R2              ; Get address of directory ASCII string
000166  017703  177614                      MOV     @NTUIC,R3               ; Get binary network UIC from SYSCM
000172  005004                              CLR     R4                      ; Suppress leading zeros, include "[,]"
000174                                      CALL    .PPASC                  ; Convert UIC to ASCII
000200  162702  001334'                     SUB     #FILDIR,R2              ; Compute length of ASCII UIC
```

```
                MOV     #WORKSP,R0              ; Get address of output event string
                MOV     #PARERR,R1              ; Copy address of $EDMSG instructions
                MOV     R3,R2                   ; Copy address of $EDMSG arguments
                CALL    $EDMSG                  ; Format the output string
                CLRB    (R0)                    ; Make it an ASCIZ string
                MOV     #WORKSP,R2              ; Recover error-specific string address
                CALL    LOGEVT                  ; Log the event
                SEC                             ; Set C-bit to indicate load error
                BR      30$                     ; Branch to common code

        20$:    MOV     R2,PCBADR               ; Save PCB address
                MOV     P.REL(R2),BLOCK         ; Copy partition mapping bias
                MOV     P.SIZE(R2),BLOCK+2      ; Copy partition size (32-word blocks)

        30$:    MOV     (SP)+,(SP)+             ; Deallocate work area but save C-bit
                MOV     (SP)+,R3                ; Retrieve address of delimiter
                MOVB    (SP)+,(R3)              ; Replace it with original data
                RETURN                          ; Exit

                .IFF                            ; If no PLAS support ...
```

```
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL   VALUE              REFERENCES

M$HIGH  = 000003            #6-83
M$3100  = 000000            #6-83
M$3101  = 000001            #6-83
M$3102  = 000002            #6-83
M$3103  = 000003            #6-83       6-83
M.KTEF  = 000002             8-202
NTUIC     000006  RG        #8-234      9-302
N$$UMR  = 000001            #4-2        6-87        7-112       8-200       9-312       9-321       9-367       9-392       10-516
                             10-553     12-657
OP$INI  = 000000            #6-83
OP$TER  = 000001            #6-83
PARHD     000010  RG        #8-235
PAR$$$  = 000027            #12-651     12-651      #14-765     14-765      #15-794     15-794      #15-796     15-796      #15-808
                             15-808
PDSPL     000024  RG        #8-241      10-547      12-680
PDVIDX    001400  R         #8-212      *9-403      10-543      11-602
PDVTA     000026  RG        #8-242      11-601
PH$HDE  = 000004            #6-83
PH$LOC  = 000002            #6-83
PH$MTS  = 000003            #6-83
PH$UMP  = 000000            #6-83
PH$WCS  = 000001            #6-83
PWRAST    001152  R          9-318      #12-648
P$$LUT  = ******             7-137
QRMVF     000012  RG        #8-236      9-361       12-670
RDBSZ     000030  RG        #8-243      10-522      10-524      10-558      10-559
RE$ADC  = 000004            #6-83
RE$ADF  = 000017            #6-83
RE$ADR  = 000007            #6-83
RE$BLK  = 000010            #6-83
RE$CAF  = 000014            #6-83
RE$DAT  = 000001            #6-83
RE$DRP  = 000016            #6-83
RE$LDT  = 000013            #6-83
RE$LSN  = 000012            #6-83
RE$NML  = 000001            #6-83
RE$OPE  = 000004            #6-83
RE$OPR  = 000000            #6-83
RE$RCV  = 000001            #6-83
RE$SED  = 000011            #6-83
RE$SKW  = 000006            #6-83
RE$STA  = 000002            #6-83
RE$SUM  = 000003            #6-83
RE$SYN  = 000000            #6-83
RE$TME  = 000021            #6-83
RE$TMO  = 000000            #6-83
RE$TMR  = 000020            #6-83
RE$UPT  = 000002            #6-83
RE$URE  = 000003            #6-83
RE$VER  = 000005            #6-83
RE$VRQ  = 000015            #6-83
RT$INI  = 000002            #6-83
```

```

                                        .IFT

                                        FSRSZ$  0,,MLDCOD               ; Define FCS work area

                                        .IFTF

                                ;       Executive Vector Table

000566                          EXEVEC:
000566  000000                          .WORD   0
000570  000000G                 ISAR5:: .WORD   KISAR5
000572  000000G                 BLXIO:: .WORD   $BLXIO
                                        .IF     DF      R$$MPL          ; If system is RSX-11M+ ...
                                NCPU::  .WORD   $NCPU
                                        .ENDC
000574  000000G                 NTUIC:: .WORD   $NTUIC
000576  000000G                 PARHD:: .WORD   $PARHD
000600  000000G                 QRMVF:: .WORD   $QRMVF
000602  000000G                 TKTCB:: .WORD   $TKTCB
000604  000000G                 CCBRT:: .WORD   $CCBRT
000606  000000G                 LDBGT:: .WORD   $LDBGT
000610  000000G                 LDBRT:: .WORD   $LDBRT
000612  000000G                 PDSPL:: .WORD   $PDSPL
000614  000000G                 PDVTA:: .WORD   $PDVTA
000616  000000G                 RDBSZ:: .WORD   $RDBSZ
000620  000000G                 SLTMA:: .WORD   $SLTMA
000622  000000G                 STDDM:: .WORD   $STDDM

        000017                  EXEVCL  = .-EXEVEC/2
```

```
                                                                                            K  1
UPDPVC - UPDATE PERMANENT VIRTU MACRO V05.03b Saturday 29-Jun-85 05:55  Page 5
LOCAL DATA

    131                                                 .SBTTL  LOCAL DATA
    132
    133                                                 .NLIST  BEX
    134 000000     011      120      126   PVCDF:  .ASCII  <11>/PVC$DF/<11>
    135         000010                                  PVCLN=.-PVCDF
    136
    137 000010  057760                         ONSTR:  .RAD50  /ON/
    138 000012  057266                                  .RAD50  /OFF/
    139
    140                                                 .LIST   BEX
    141                                                 .EVEN
    142
```

```
                                                                                            L  1
UPDPVC - UPDATE PERMANENT VIRTU MACRO V05.03b Saturday 29-Jun-85 05:55  Page 6
$UPPVC - UPDATE PERMANENT VIRTUAL CIRCUIT
```

```
UPDRDT - UPDATE REMOTE DTE DEFI MACRO V05.03b Saturday 29-Jun-85 05:55  Page 6-1
Symbol table

C.ADR   000010          RDIDF   000000R         $OBUF = ****** GX       $UPRDT  000010RG        .COMMA= ****** GX
C.LEN = 000022          RDTLN = 000010          $OFDB = ****** GX       $WDATA= ****** GX       .FLAGS= ****** GX
C.NAM   000004          $BUFI = ****** GX       $PTR50= ****** GX       .BUFF = ****** GX       .TEMP = ****** GX
F.NRBD= ****** GX       $MVASC= ****** GX

. ABS.  000022    000   (RW,I,GBL,ABS,OVR)
        000142    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8830 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:05.07
SY:UPDRDT.V2,[132,134]UPDRDT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDRDT
```

```
UPDRDT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 1
```

```
                                                .SBTTL  UPDLL - UPDATE DOWNLINE LOAD DATABASE
                                        ;+
                                        ; **-UPDLL-UPDATE DOWNLINE LOAD DATABASE
                                        ;
                                        ; INPUTS:       $TMLST  = ADDRESS OF TEMPLATE LIST
                                        ;               TEMP    = REM$DF TEMPLATE ADDRESS
                                        ;
                                        ; OUTPUT:       DOWNLINE LOAD DATABASE, IF ANY, IS
                                        ;               WRITTEN INTO OUTPUT FILE.
                                        ;
                                        ; REGISTERS:    ALL REGISTERS ARE PRESERVED.
                                        ;
                                        ;-

000476                                  UPDLL:
000476  004567  000000G                         JSR     R5,$SAVRG           ; SAVE R3-R5
000502  012705  000000G                         MOV     #$TMLST,R5          ; POINT TO TEMPLATE LISTHEAD
000506  016704  177266                          MOV     TEMP,R4             ; POINT TO REMOTE TEMPLATE

000512  000241                          10$:    CLC
000514  011505                                  MOV     (R5),R5             ; GET ADDRESS OF NEXT TEMPLATE
000516  001422                                  BEQ     20$                 ; IF EQ, NO DLL INFO FOR REMOTE
000520  122765  000000G 000000G                 CMPB    #CS.SER,C.STS(R5)   ; IS THIS A REMOTE DEFINITION
000526  001371                                  BNE     10$                 ; IF NE, NO .. KEEP LOOKING
000530  026564  000006  000010                  CMP     S$ADD(R5),C.ADD(R4) ; FOR THIS REMOTE NODE ?
000536  001365                                  BNE     10$                 ; NO
                                        ;
                                        ; FOUND A TEMPLATE
                                        ;
000540  004767  000000G                         CALL    $UPSER              ; UPDATE SER$DF
000544  103407                                  BCS     20$                 ; IF CS, ERROR
000546  011505                                  MOV     (R5),R5             ; POINT TO SUCCESSOR TEMPLATE
000550  122765  000000G 000000G                 CMPB    #CS.FIL,C.STS(R5)   ; IS THIS A FIL$DF TEMPLATE ?
000556  001003                                  BNE     101$                ; IF NE, INTERNAL ERROR !
000560  004767  000000G                         CALL    $UPFIL              ; UPDATE FIL$DF
000564  000207                          20$:    RETURN

000566  000167  000000G                 101$:   JMP     .CONER              ; BIG TROUBLE

        000001                                  .END
```

```
                                                                                   K  4
UPDROU - CFE UPDATE ROUTING DEF MACRO V05.03b Saturday 29-Jun-85 05:56  Page 7-1
Symbol table

CS.MOD= ****** GX       C.NBEA  000016          $ANY   = 000020         $EXIT  = 000000         $STRNG= 000004
CS.ROU= ****** GX       C.RTM   000012          $BLANK= 000006          $FAIL  = 177777         $SUBXP= 000010
C.ACST  000022          C.STS = ****** GX       $BUFI  = ****** GX      $LAMDA= 000000          $TMLST= ****** GX
C.ADD   000004          EXIT    000104R     002 $CLNUP= ****** GX       $MOVPT= ****** GX       $UPROU  000002RG
C.AHOP  000020          F.NRBD= ****** GX       $CNVD  = ****** GX      $MOVST= ****** GX       $$$FLG= 177777
C.CST   000006          ROUKTB  000000RG    003 $DIGIT= 000024          $NUMBR= 000002          $$$KEY= 177777
C.HOP   000010          ROUSTB  000000RG    002 $DNUMB= 000014          $OBUF  = ****** GX      $$$STA= 000000
C.LEN = 000026          SETNUM  000122R         $EOS   = 000012         $OFDB  = ****** GX      .CONER= ****** GX
C.L2TM  000024          TEMP    000000R         $ERROR= ****** GX       $RAD50= 000016          .TPARS= ****** GX
C.NA    000014          $ALPHA= 000022

. ABS.  000026    000   (RW,I,GBL,ABS,OVR)
        000152    001   (RW,I,LCL,REL,CON)
$STATE  000112    002   (RW,D,LCL,REL,CON)
$KTAB   000000    003   (RW,D,LCL,REL,CON)
$KSTR   000000    004   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10406 Words  ( 41 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:25.45
SY:UPDROU.V2,[132,134]UPDROU/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDROU
```

```
                                                          L  4
UPDROU     CREATED BY  MACRO  ON 29-JUN-85 AT 05:56      PAGE 1
```

```
Symbol table

F.NRBD= ****** GX       SF.ADD= 000001          SF.PSS= 000040          S$HST   000010          $OFDB = ****** GX
PH3TXT  000010R         SF.CIR= 000010          S$ADD   000006          S$LEN = 000056          $SAVRG= ****** GX
PUTFLG  000510R         SF.DEV= 000020          S$CIR   000022          S$PSS   000034          $UPSER  000020RG
PUTNOD  000266R         SF.DPA= 000100          S$DEV   000032          $CBTA3= ****** GX       $WDATA= ****** GX
PUTOCT  000442R         SF.DPC= 000200          S$DPA   000050          $CNVD = ****** GX       .BUFF = ****** GX
PUTR50  000366R         SF.HAD= 000004          S$DPC   000054          $C5TA1= ****** GX       .FLAGS= ****** GX
SERDF   000000R         SF.HST= 000002          S$FLG   000004          $MVASC= ****** GX       .TEMP = ****** GX
SERLN = 000010          SF.PH3= 000400          S$HAD   000012          $OBUF = ****** GX

. ABS.  000056    000   (RW,I,GBL,ABS,OVR)
        000560    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 218 Words  ( 1 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:07.46
SY:UPDSER.V2,[132,134]UPDSER/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDSER
```

```
    000520  001401                           BEQ     80$             ; IF EQ, NO
    000522  105740                           TSTB    -(R0)           ; BACK UP OVER LAST "!"
    000524  112720  000054           80$:    MOVB    #<',>,(R0)+     ; INSERT DELIMETER
    000530  010067  000000G                  MOV     R0,$OBUF        ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
    000534  012604                           MOV     (SP)+,R4        ; RESTORE R4
    000536  000207                           RETURN                  ; RETURN TO CALLER
    103                                 ;****
    104                                 ; INITIALIZE TABLE GENERATION
    105                                 ;****
    106
    107 000036                                  ISTAT$  STASTB,STAKTB
    108
    109 000036                                  STATE$
    110 000036                                   TRAN$  $NUMBR,,SETADD
    111
    112 000036                                  STATE$
    113 000036                                   TRAN$  <',>,,$MOVPT
    114
    115 000036                                  STATE$
    116 000036                                   TRAN$  $RAD50,,$MOVPT
    117
    118 000036                                  STATE$
    119 000036                                   TRAN$  <',>,SFLAG,$MOVPT
    120 000036                                   TRAN$  $LAMDA,EXIT
    121
    122 000036                                  STATE$  SFLAG
    123 000036                                   TRAN$  !SFS,STACST             ; SKIP FLAG BITS
    124 000036                                   TRAN$  <',>,STACST,SETFLG
    125 000036                                   TRAN$  $LAMDA,EXIT
    126
    127 000036                                  STATE$  STACST                  ; SET STAION COST
    128 000036                                   TRAN$  $NUMBR,,SETCST
    129
    130 000036                                  STATE$                          ; ACTIVE POLLING RATIO
    131 000036                                   TRAN$  <',>,APRNUM,$MOVPT
    132 000036                                   TRAN$  $LAMDA,EXIT
    133
    134 000036                                  STATE$  APRNUM
    135 000036                                   TRAN$  $NUMBR,,SETAPR
    136
    137 000036                                  STATE$                          ; HELLO TIMER
    138 000036                                   TRAN$  $LAMDA,EXIT,SETHTM
    139
    140 000036                                  STATE$  EXIT
    141 000036                                   TRAN$  $LAMDA,$EXIT,$CLNUP
    142
    143                                 ;
    144                                 ; LOOK FOR STATION FLAGS
    145                                 ;
    146 000036                                  STATE$  SFS,
    147 000036                                   TRAN$  "SF"
    148
    149 000036                                  STATE$
    150 000036                                   TRAN$  '.,BITS
    151
    152 000036                                  STATE$  SKIP
    153 000036                                   TRAN$  <',>,$EXIT,SETFLG
    154 000036                                   TRAN$  '!,SFS
    155
    156 000036                                  STATE$  BITS
    157 000036                                   TRAN$  $RAD50,SKIP
    158
    159 000036                                  STATE$
```

```
        .TITLE  UPDSVC - UPDATE SWITCHED VIRTUAL CIRCUIT
        .IDENT  /V05.00/
        .ENABL  LC

;
; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       UPDATE SWITCHED VIRTUAL CIRCUIT
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-AUG-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;



```

```
    120                                         ;****
    121                                         ; INITIALIZE TABLE GENERATION
    122                                         ;****
    123
    124 000000                                          ISTAT$  UNTSTB,UNTKTB
    125
    126 000000                                          STATE$
    127 000000                                           TRAN$  $NUMBR,,SETUNT
    128
    129 000000                                          STATE$
    130 000000                                           TRAN$  <',>,,$MOVPT
    131
    132 000000                                          STATE$
    133 000000                                           TRAN$  $NUMBR,,SETCW0
    134
    135 000000                                          STATE$
    136 000000                                           TRAN$  <',>,,$MOVPT
    137
    138 000000                                          STATE$
    139 000000                                           TRAN$  $NUMBR,,SETCW1
    140
    141                                                                                ; OPTIONAL PARAMETERS
    142 000000                                          STATE$                          ; SECONDARY CSR
    143 000000                                           TRAN$  $LAMDA,,SETCSR
    144 000000                                          STATE$                          ; CIRCUIT COST
    145 000000                                           TRAN$  $LAMDA,,SETCST
    146 000000                                          STATE$                          ; DEAD POLLING RATIO
    147 000000                                           TRAN$  $LAMDA,,SETDPR
    148 000000                                          STATE$                          ; PROTOCOL EMULATOR CHARACTERISTICS
    149 000000                                           TRAN$  $LAMDA,$EXIT,SETPCH
    150
    151 000000                                          STATE$
```

```
**FILE**ID**UPDX2P
```

```
UU      UU   PPPPPPPP     DDDDDDDD     XX      XX     222222     PPPPPPPP
UU      UU   PPPPPPPP     DDDDDDDD     XX      XX     222222     PPPPPPPP
UU      UU   PP      PP   DD      DD   XX      XX   22      22   PP      PP
UU      UU   PP      PP   DD      DD   XX      XX   22      22   PP      PP
UU      UU   PP      PP   DD      DD     XX  XX             22   PP      PP
UU      UU   PP      PP   DD      DD     XX  XX             22   PP      PP
UU      UU   PPPPPPPP     DD      DD       XX             22     PPPPPPPP
UU      UU   PPPPPPPP     DD      DD       XX             22     PPPPPPPP
UU      UU   PP           DD      DD     XX  XX         22       PP
UU      UU   PP           DD      DD     XX  XX         22       PP
UU      UU   PP           DD      DD   XX      XX     22         PP              ....
UU      UU   PP           DD      DD   XX      XX     22         PP              ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX   2222222222   PP              ....
UUUUUUUUUU   PP           DDDDDDDD     XX      XX   2222222222   PP              ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
UPDX2P - UPDATE X.25 LEVEL 2 CH MACRO V05.03b Saturday 29-Jun-85 05:58
Table of contents
```

```
     57                                         .SBTTL  MACRO CALLS AND LOCAL DATA
     58                                 ;
     59                                 ; LOCAL MACROS
     60                                 ;
     61
     62                                         .MACRO  PRINT   TEXT
     63                                         MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
     64                                         CALL    $PRINT          ; PRINT MESSAGE
     65                                         .ENDM   PRINT
     66
     67                                         .MACRO  ERROR$  TEXT
     68                                 ;
     69                                 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
     70                                 ; MESSAGE STRING.
     71                                 ;
     72                                         .IF DIF <TEXT><R0>
     73                                         MOV     TEXT,R0                         ; GET ADDRESS OF ERROR MESSAGE
     74                                         .ENDC
     75                                         CALL    $CFERR                          ; PRINT ERROR MESSAGE
     76                                         .ENDM   ERROR$
     77
     78                                         .MACRO  EPRINT  TEXT
     79                                 ;
     80                                 ; PRINT TEXT ON ERROR LUN
     81                                 ;
     82                                         .IF DIF <TEXT><R0>
     83                                         MOV     TEXT,R0                         ; GET ADDRESS OF ASCIZ STRING TO PRINT
     84                                         .ENDC
     85                                         CALL    $CFER1                          ; PRINT MESSAGE ON ERROR LUN
     86                                         .ENDM   EPRINT
     87                                 ;
     88                                 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
     89                                 ;
     90                                         .MACRO  $$$PTN  ROU
     91                                         .MACRO  'ROU    OFS,OCTL,OPT,?L
     92                                         .IF NB <OPT>
     93                                         MOV     OPT,R2
     94                                         CALL    $PTOPT
     95                                         BCS     L
     96                                         .IFTF
     97                                         MOV     OFS,R2
     98                                         .IF NB <OCTL>
     99                                         MOV     SP,R1
    100                                         .IFF
    101                                         CLR     R1
    102                                         .ENDC ; IF NB <OCTL>
    103                                         CALL    'ROU
    104                                         .IFT  ; IF NB <OPT>
    105                                 L:
    106                                         .ENDC ; IF NB <OPT>
    107                                         .ENDM ; 'ROU
    108                                         .ENDM ; $$$PTN
    109
    110 000000                                  $$$PTN  $PTNUM
    111 000000                                  $$$PTN  $PTNMB
    112                                 ;
    113                                 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
```

```
                                                                                    K 12
UPDX3P - UPDATE X.25 LEVEL 3 CH MACRO V05.03b Saturday 29-Jun-85 05:59  Page 5-1
Symbol table

C.CT     000012          C.MW     000011          F.NRBD= ****** GX        $OBUF  = ****** GX        $WDATA= ****** GX
C.DFP    000004          C.MXP    000006          X3PDF    000000R         $OFDB  = ****** GX        .BUFF  = ****** GX
C.DW     000010          C.RM     000017          X3PLN =  000010          $PTNMB= ****** GX         .COMMA= ****** GX
C.KM     000016          C.RT     000014          $BUFI = ****** GX        $PTNUM= ****** GX         .FLAGS= ****** GX
C.KT     000013          C.SM     000020          $MVASC= ****** GX        $UPX3P   000010RG         .TEMP  = ****** GX
C.LEN =  000021          C.ST     000015

. ABS.   000021     000    (RW,I,GBL,ABS,OVR)
         000270     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8783 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.03
SY:UPDX3P.V2,[132,134]UPDX3P/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDX3P
```

```
                                                                                    L 12
UPDX3P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59       PAGE 1
```

```
    306 000204  010267  001316'                 MOV     R2,DSDS+4               ; Move length to dataset descriptor
    307
    308                                         .IFTF                           ; For any operating system ..
    309
    310 000210  005067  001376'                 CLR     CURQST                  ; No load request outstanding
    311
    312                                         .IF     DF      N$$UMR          ; If system has UMRs ...
    313
    314 000214  005067  001364'                 CLR     LDBPTR                  ; No large data buffer yet
    315
    316                                         .ENDC                           ; End UMR conditional
    317
    318 000220                                  SPRA$S  #PWRAST                 ; Specify AST for powerfail recovery
    319 000232                                  SREA$S  #ABOAST                 ; Specify AST for requested aborts
    320
    321                                         .IF     DF      N$$UMR          ; If system has UMRs ...
    322
    323
    324                                 ;       In order to guarantee that devices which DMA in their own microcode
    325                                 ;       can access the microcode, allocate a large data buffer before
    326                                 ;       continuing.  LDBs are always UMR-mapped if the system has UMRs.
    327
    328
    329 000244                          5$:     CALL    $SWSTK,7$               ;; Switch to system state
    330 000250                                  CALL    @LDBGT                  ;; Allocate a UMR-mapped buffer
    331 000254  103413                          BCS     6$                      ;; Branch if failure
    332 000256  010467  001364'                 MOV     R4,LDBPTR               ;; Save address of CCB/LDB
    333 000262  016467  000014  001366'         MOV     C.BUF1(R4),LDBIAS       ;; Save LBD's mapping bias
    334 000270  016467  000016  001370'         MOV     C.BUF1+2(R4),LDBADR     ;; Save LDB's starting address
    335 000276  012764  001026  000010          MOV     #FC.MLD+FS.WLB,C.FNC(R4);; LDB used only for writes
    336
    337 000304                          6$:     RETURN                          ;; Return to user state at "7$"
    338 000306  005767  001364'         7$:     TST     LDBPTR                  ;  Was an LDB allocated?
    339 000312  001007                          BNE     10$                     ;  Branch if so
    340 000314                                  DIR$    #SETTMR                 ;  Set a timer
    341 000322                                  DIR$    #WAIT                   ;  Wait for timer to expire
    342 000330  000745                          BR      5$                      ;  Attempt LDB allocation again
    343
    344                                         .ENDC                           ;  End UMR conditional
    345
    346
    347                                 ;       Enter system state to dequeue a CCB from our TCB receive list.
    348                                 ;       Each CCB is a load request queued to MLD by AUX on behalf of
    349                                 ;       a DECnet device driver.
    350
    351
    352 000332                          10$:    CALL    $SWSTK,50$              ;; Switch to system state
    353 000336  016704  001376'         15$:    MOV     CURQST,R4               ;; Get current request CCB
    354 000342  001404                          BEQ     20$                     ;; Branch if none there
    355 000344                                  CALL    @CCBRT                  ;; Return CCB to its pool
    356 000350  005067  001376'                 CLR     CURQST                  ;; Request no longer outstanding
    357
    358 000354  017700  177434          20$:    MOV     @TKTCB,R0               ;; Get address of TCB
    359 000360  062700  000012                  ADD     #T.RCVL,R0              ;; Advance to TCB receive listhead
    360
    361 000364                          30$:    CALL    @QRMVF                  ;; Dequeue a load request CCB
    362 000370  103407                          BCS     40$                     ;; Branch if no CCB(s) queued
```

```
                .SBTTL  SRNAM   - Search For Named Partition
;+
;                                       *** SRNAM ***
;
; Function:     To search the system PCB list for a named partition.  This
;               routine is supplied for systems without PLAS support.
;
; Call Format:  CALL    SRNAM
;
; Inputs:       R3 = address of RAD50 doubleword partition name
;
; Outputs:      C-bit set:      partition not found
;               C-bit clear:    partition found, R2 = PCB address
;
; Registers:    R2 destroyed
;
;-

SRNAM:  MOV     @PARHD,R2               ; Get address of first PCB
10$:    CMP     P.NAM(R2),(R3)          ; Match on first word of name?
        BNE     20$                     ; Branch if not
        CMP     P.NAM+2(R2),2(R3)       ; Match on second word of name?
        BEQ     50$                     ; Branch if so

20$:    BIT     #PS.SYS,P.STAT(R2)      ; System-controlled partition?
        BEQ     40$                     ; Branch if not
        TST     P.SUB(R2)               ; Is there another partition?
        BEQ     30$                     ; Branch if not
        MOV     P.SUB(R2),R2            ; Get next subpartition PCB address
        BR      10$                     ; Look for match

30$:    MOV     P.MAIN(R2),R2           ; Get address of main partition PCB
40$:    MOV     (R2),R2                 ; Get address of next PCB
        BNE     10$                     ; Branch if one there
        SEC                             ; Set C-bit for failure

50$:    RETURN                          ; Exit

        .ENDC                           ; End PLAS conditional
```

```
                                                                   K 15
MLD       CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 7
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

RT$OFF  = 000001           #6-83
RT$ON   = 000000           #6-83
R$$MPL  = ******             6-54        8-186       8-231       9-273       9-407
R$$11D  = ******             6-84
R$$11M  = 000000             6-84
R$$11S  = ******             6-62        6-84
SC$OFF  = 000001           #6-83
SC$ON   = 000000           #6-83
SC$RST  = 000003           #6-83
SC$SHU  = 000002           #6-83
SETTMR    001346 R         #8-202        9-340
SF.ACT  = 000200           #6-85
SF.ENA  = 000100           #6-85
SF.LPB  = 000004           #6-85
SF.MFL  = 000040           #6-85
SF.PAC  = 000020           #6-85
SF.REA  = 000010           #6-85
SF.SER  = 000001           #6-85
SF.SVC  = 000002           #6-85
SF.UNL  = 000040           #6-85
SLTMA     000032 RG        #8-244        9-400
STDDM     000034 RG        #8-245        12-674
SV$DUM  = 000001           #6-83
SV$LOA  = 000000           #6-83
S.BFHD  = 000020             8-221       8-221
S.COST    000001           #6-85
S.FDB   = 000140             8-166       8-166
S.FLG     000000           #6-85
S.LEN     000004           #6-85
S.NMST    000002           #6-85
S.OWNR    000003           #6-85
S.TSEF  = 000002             8-203
TKTCB     000014 RG        #8-237        9-284       9-358       12-668
TMREFL  = 000003           #7-115        8-202       8-203
T.LDV     000044             9-285
T.RCVL    000012             9-359       12-669
U.DCB     000000             9-286
WAIT      001360 R         #8-203        9-341
WORKSP    001406 R         #8-215        13-719      13-723
X$$MCB  = ******             6-84        6-84
ZF.COU  = 001000           #6-84
ZF.DDM  = 000001           #6-84
ZF.DIA  = 004000           #6-84
ZF.DLC  = 000002           #6-84
ZF.DVP  = 100000           #6-84
ZF.INI  = 040000           #6-84
ZF.KMX  = 000020           #6-84
ZF.LLC  = 000004           #6-84
ZF.LMC  = 000100           #6-84
ZF.MAN  = 020000           #6-84
ZF.MFL  = 000010           #6-84
ZF.MTM  = 000400           #6-84
```

```
                                                                   L 15
MLD       CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 8
```

```
                                        .SBTTL  MLD     - Microcode Loader Main Routine

000000                                  .PSECT  MLDCOD,I,RO,REL,OVR,LCL
                                ;+
                                ;                         *** MLD ***
                                ;
                                ; Function:     This program is used service requests from DECnet device
                                ;               drivers for specific microcode files (RSX-11M) or partitions
                                ;               (RSX-11S).
                                ;
                                ; Inputs:       No explicit inputs.  The task's receive listhead in the TCB
                                ;               should be non-empty when the task begins execution.  Load
                                ;               requests are queued to this list by AUX upon request from
                                ;               device drivers.
                                ;
                                ; Outputs:      No explicit outputs.  MLD will perform a single load session
                                ;               with each driver that has made a request.  Requests are
                                ;               handled FIFO with no multiplexing.
                                ;-

000000                          MLD:    ALUN$S  #EVTLUN,#"NM,#0         ; Assign LUN to network management

000020                          1$:     .IFT                            ; If system is RSX-11M ...

                                        .IF     DF R$$MPL               ; If system is RSX-11M-PLUS
                                        .IF     NDF R$$PRO              ; And not P/OS

                                        DIR$    #GINVEC                 ; Initialize executive vector table
                                        BCC     2$                      ; Branch if no error
                                        CALL    ABOAST                  ; Else , clean up and exit
                                2$:
                                        .ENDC   ; NDF R$$PRO
                                        .ENDC   ; DF R$$MPL

                                        FINIT$                          ; Initialize FCS impure area
                                        MOV     @TKTCB,R0               ; Get our TCB address
                                        MOV     T.LDV(R0),R0            ; Get our load device UCB address
                                        MOV     U.DCB(R0),R3            ; Get load device DCB address
                                        MOV     D.NAM(R3),FILDEV        ; Copy load device name
                                        SUB     D.UCB(R3),R0            ; Compute relative UCB address
                                        MOV     D.UCBL(R3),R1           ; Get UCB length
                                        CALL    $DIV                    ; Compute relative unit number
                                        ADD     D.UNIT(R3),R0           ; Compute absolute unit number
                                        BIC     #177400,R0              ; Clear high byte
                                        MOV     R0,R1                   ; Copy it
                                        MOV     #FILDEV+2,R0            ; Get address of ASCII unit number
                                        MOV     #CVTPRM,R2              ; Get conversion parameters
                                        CALL    $CBTA                   ; Convert unit to ASCII
                                        MOVB    #':,(R0)+               ; Complete the device name
                                        SUB     #FILDEV,R0              ; Get length of device name
                                        MOV     R0,DSDS                 ; Move it to dataset descriptor

                                        MOV     #FILDIR,R2              ; Get address of directory ASCII string
                                        MOV     @NTUIC,R3               ; Get binary network UIC from SYSCM
                                        CLR     R4                      ; Suppress leading zeros, include "[,]"
                                        CALL    .PPASC                  ; Convert UIC to ASCII
                                        SUB     #FILDIR,R2              ; Compute length of ASCII UIC
```

```
                                                 .SBTTL  $UPPVC - UPDATE PERMANENT VIRTUAL CIRCUIT
                                        ;+
                                        ; *** - $UPPVC - UPDATE PERMANENT VIRTUAL CIRCUIT
                                        ;
                                        ; INPUT:
                                        ;          R5 - TEMPLATE ADDRESS
                                        ;
                                        ; OUTPUT:
                                        ;          C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;
                                        ;-

000014                                  $UPPVC::
000014  012767  000000G 000000C                  MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
000022  010567  000000G                          MOV     R5,.TEMP         ; SAVE TEMPLATE ADDRESS
000026  005725                                   TST     (R5)+            ; POINT TO C.STS
000030  010567  000000G                          MOV     R5,.FLAGS        ; SAVE ADDRESS OF FLAGS WORD
000034  012767  177777  000000G                  MOV     #-1,.COMMA       ; INITIALIZE FOR OPTIONS
000042  012700  000000G                          MOV     #.BUFF,R0        ; POINT AT START OF OUTPUT BUFFER
000046  012701  000000'                          MOV     #PVCDF,R1        ; POINT AT START OF PVC DEFINITION
000052  012702  000010                           MOV     #PVCLN,R2        ; GET LENGTH OF PVC DEFINITION
000056  004767  000000G                          CALL    $MVASC           ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
000062  010067  000000G                          MOV     R0,$OBUF         ; SAVE CURRENT POSITION IN OUTPUT BUFFER
000066  026527  177776G 015355                   CMP     P$VOWN-2(R5),#^RDLM ; IS THIS A DLM PVC ?
000074  001033                                   BNE     10$              ; IF NE, NO
000076  162767  000002  000000G                  SUB     #2,.COMMA        ; SUPPRESS FIRST 3 COMMAS
000104                                           $PTR50  #P$VNAM,#1       ; OUTPUT DEVICE NAME
000120  112777  000055  000000G                  MOVB    #'-,@$OBUF       ; DELIMIT CONTROLLER NUMBER
000126  005267  000000G                          INC     $OBUF            ; ADVANCE BUFFER POINTER
000132                                           $PTNMB  #P$VNAM+2        ; CONTROLLER NUMBER
000144                                           $PTNMB  #P$VNAM+3        ; TRIB NUMBER
000156  005367  000000G                          DEC     $OBUF            ; BACK OVER DECIMAL POINT
000162  000406                                   BR      20$              ; JOIN COMMON CODE
000164                                  10$:     $PTR50  #P$VNAM,#2       ; NAME
000200                                  20$:     $PTNUM  #P$VLCN          ; CHANNEL NUMBER
000212                                           $PTNUM  #P$VCT           ; COUNTER TIMER
000224  004767  000152                           CALL    SETSTA           ; STATE
000230                                           $PTNUM  #P$VPSZ,,#PV$PSZ         ; BLOCK SIZE
000254                                           $PTNUM  #P$VWSZ,,#PV$WSZ         ; WINDOW SIZE
000300                                           $PTR50  #P$VOWN,#1,#PV$OWN ; OWNER
000326                                           $PTNMB  #P$VFLG,OCTAL,#CS.FLG ; FLAGS
000352  016701  000000G                          MOV     $OBUF,R1         ; CALCULATE LENGTH OF BUFFER
000356  162701  000000G                          SUB     #.BUFF,R1        ;  ...
000362  010167  000000C                          MOV     R1,$OFDB+F.NRBD  ; SET LENGTH OF BUFFER
000366  004767  000000G                          CALL    $WDATA           ; WRITE THE RECORD TO THE OUTPUT FILE
000372  012767  000000G 000000C                  MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
000400  000207                                   RETURN
```

```
UPDRDT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

C.ADR     000010         #5-140      6-174
C.LEN   = 000022         #5-141
C.NAM     000004         #5-139      6-173
F.NRBD  = ******  GX     *6-163     *6-177      *6-179
RDTDF     000000 R       #5-144      5-145       6-169
RDTLN   = 000010         #5-145      6-170
$BUFI   = ******  GX      6-179
$MVASC  = ******  GX      6-171
$OBUF   = ******  GX     *6-172      6-175
$OFDB   = ******  GX     *6-163     *6-177      *6-179
$PTR50  = ******  GX      6-173      6-174
$UPRDT    000010 RG      #6-162
$WDATA  = ******  GX      6-178
.BUFF   = ******  GX      6-163      6-168       6-176
.COMMA  = ******  GX     *6-167
.FLAGS  = ******  GX     *6-166
.TEMP   = ******  GX     *6-164
```

```
UPDRDT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 2
```

```
                                                              L 3
UPDREM - CFE UPDATE REMOTE DEFI MACRO V05.03b Saturday 29-Jun-85 05:55  Page 10-1
Symbol table

ADD2     000026R      002 REMDF    000023R          S$FLG    000004          $DIGIT= 000024          $SAVRG= ****** GX
BUFF     000033R          REMKTB   000000RG     003 S$HAD    000012          $DNUMB= 000014          $STRNG= 000004
CS.FIL= ****** GX         REMLEN=  000020           S$HST    000010          $EOS   = 000012         $SUBXP= 000010
CS.REM= ****** GX         REMLN =  000010           S$LEN =  000056          $ERROR= ****** GX       $TMLST= ****** GX
CS.SER= ****** GX         REMSTB   000000RG     002 S$PSS    000034          $EXIT  = 000000         $UPFIL= ****** GX
C.ADD    000010           SETADD   000444R          TEMP     000000R         $FAIL  = 177777         $UPREM  000154RG
C.LEN =  000012           SETARE   000404R          UPDLL    000476R         $LAMDA= 000000          $UPSER= ****** GX
C.NAM    000004           SETNAM   000342R          $ALPHA=  000022          $MOVPT= ****** GX       $WDATA= ****** GX
C.STS =  ****** GX        S$ADD    000006           $ANY   = 000020          $MVASC= ****** GX       $$$FLG= 177777
EXIT     000036R      002 S$CIR    000022           $BLANK=  000006          $NUMBR= 000002          $$$KEY= 177777
F.NRBD= ****** GX         S$DEV    000032           $BUFI  = ****** GX       $OBUF  = ****** GX      $$$STA= 000000
LA     = 000074           S$DPA    000050           $CNVD  = ****** GX       $OFDB  = ****** GX      .CONER= ****** GX
RA     = 000076           S$DPC    000054           $C5TA  = ****** GX       $RAD50= 000016          .TPARS= ****** GX
REMDEF   000002R

. ABS.   000056      000   (RW,I,GBL,ABS,OVR)
         000572      001   (RW,I,LCL,REL,CON)
$STATE   000042      002   (RW,D,LCL,REL,CON)
$KTAB    000000      003   (RW,D,LCL,REL,CON)
$KSTR    000000      004   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 10456 Words  ( 41 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:17.13
SY:UPDREM.V2,[132,134]UPDREM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDREM
```

```
                                                  M 3
UPDREM     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55     PAGE 1
```

```
                                                                           L  4
UPDROU      CREATED BY  MACRO  ON 29-JUN-85 AT 05:56      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE               REFERENCES

CS.MOD  = ******  GX         6-184
CS.ROU  = ******  GX         6-177
C.ACST    000022            #4-89
C.ADD     000004            #4-82        6-188
C.AHOP    000020            #4-88
C.CST     000006            #4-83
C.HOP     000010            #4-84
C.LEN   = 000026            #4-91
C.L2TM    000024            #4-90
C.NA      000014            #4-86
C.NBEA    000016            #4-87
C.RTM     000012            #4-85
C.STS   = ******  GX         6-177       6-184
F.NRBD  = ******  GX        *6-186
ROUKTB    000000 RG         #5-100       6-192
ROUSTB    000000 RG         #5-100       6-193
SETNUM    000122 R          #7-219
TEMP      000000 R          #4-73       *6-183      *6-188       7-221      *7-223
$ALPHA  = 000022            #5-100
$ANY    = 000020            #5-100
$BLANK  = 000006            #5-100
$BUFI   = ******  GX         6-186
$CNVD   = ******  GX         7-224
$DIGIT  = 000024            #5-100
$DNUMB  = 000014            #5-100
$EOS    = 000012            #5-100
$ERROR  = ******  GX        *6-196
$EXIT   = 000000            #5-100
$FAIL   = 177777            #5-100
$GPRM   = ******             5-100
$LAMDA  = 000000            #5-100
$MOVST  = ******  GX         6-189
$NUMBR  = 000002            #5-100
$OBUF   = ******  GX         7-220      *7-225
$OFDB   = ******  GX        *6-186
$RAD50  = 000016            #5-100
$RONLY  = ******             5-100       5-100       5-100
$STRNG  = 000004            #5-100
$SUBXP  = 000010            #5-100
$TMLST  = ******  GX         6-173
$UPROU    000002 RG         #6-172
$$$FLG  = 177777            #5-100
$$$KEY  = 177777            #5-100
.CONER  = ******  GX         6-202
.TPARS  = ******  GX         6-194
```

```
                                                                           M  4
UPDROU      CREATED BY  MACRO  ON 29-JUN-85 AT 05:56      PAGE 2
```

```
                                                                  L  5
UPDSER     CREATED BY  MACRO  ON 29-JUN-85 AT 05:56     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE            REFERENCES

F.NRBD  = ******  GX      5-107      *5-108      *5-152      *5-154
PH3TXT    000010 R       #4-57        5-143
PUTFLG    000510 R        5-142      #9-263
PUTNOD    000266 R        5-118       5-121      #6-171
PUTOCT    000442 R        5-133       5-136      #8-233
PUTR50    000366 R        5-124       5-127       5-130       5-139      #7-206
SERDF     000000 R       #4-55        4-56        5-113
SERLN   = 000010         #4-56        5-114
SF.ADD  = 000001         #4-82        5-119
SF.CIR  = 000010         #4-85        5-125
SF.DEV  = 000020         #4-86        5-128
SF.DPA  = 000100         #4-88        5-134
SF.DPC  = 000200         #4-89        5-137
SF.HAD  = 000004         #4-84        5-140
SF.HST  = 000002         #4-83        5-122
SF.PH3  = 000400         #4-90        5-143
SF.PSS  = 000040         #4-87        5-131
S$ADD     000006         #4-69        5-119
S$CIR     000022         #4-72        5-125
S$DEV     000032         #4-73        5-128
S$DPA     000050         #4-75        5-134
S$DPC     000054         #4-76        5-137
S$FLG     000004         #4-68        5-110
S$HAD     000012         #4-71        5-140
S$HST     000010         #4-70        5-122
S$LEN   = 000056         #4-77
S$PSS     000034         #4-74        5-131
$CBTA3  = ******  GX      8-241
$CNVD   = ******  GX      6-185       6-189
$C5TA1  = ******  GX      7-215
$MVASC  = ******  GX      5-115
$OBUF   = ******  GX     *5-116       5-146       6-172      *6-191       7-211      *7-218       8-234      *8-244       9-263
                         *9-277
$OFDB   = ******  GX      5-107      *5-108      *5-152      *5-154
$SAVRG  = ******  GX      5-106
$UPSER    000020 RG      #5-105
$WDATA  = ******  GX      5-153
.BUFF   = ******  GX      5-108       5-112       5-151
.FLAGS  = ******  GX     *5-111       6-176       7-212       8-239       9-266
.TEMP   = ******  GX     *5-109       6-173       7-208       8-236
```

```
                                                                  M  5
UPDSER     CREATED BY  MACRO  ON 29-JUN-85 AT 05:56     PAGE 2
```

```
                                                .SBTTL  SETCNT - CHECK THE CONTROLLER NUMBER
                                        ;+
                                        ; *** - SETCNT - CHECK THE CONTROLLER NUMBER
                                        ;
                                        ; INPUT:
                                        ;       .PNUMB = CONTROLLER NUMBER
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000540                                  SETCNT:
000540  004767  000000G                         CALL    $MOVPT          ; MOVE THE CONTROLLER NUMBER INTO THE OUTPUT BUFFER
000544  016700  000000G                         MOV     .TEMP,R0        ; GET START OF TEMPLATE
000550  126760  000000G 000010                  CMPB    .PNUMB,C.CNT(R0) ; IS THIS THE RIGHT TEMPLATE ?
000556  001402                                  BEQ     10$             ; IF EQ, YES .. OKAY
000560  062716  000002                          ADD     #2,(SP)         ; ELSE, REJECT THE TRANSITION
000564  000207                          10$:    RETURN
```

```
                                                .SBTTL  $UPSTA - UPDATE STATION DEFINITION
                                        ;+
                                        ; *** - $UPSTA - UPDATE STATION DEFINITION
                                        ;
                                        ; INPUT:
                                        ;       R3-R5   - TPARS REGISTERS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000036                                  $UPSTA::
000036  012700  000000G                         MOV     #$TMLST,R0              ; POINT AT START OF TEMPLATES
000042  010001                          10$:    MOV     R0,R1                   ; SAVE CURRENT TEMPLATE ADDRESS
000044  011000                                  MOV     (R0),R0                 ; GET NEXT TEMPLATE
000046  001436                                  BEQ     101$                    ; IF EQ, NOT FOUND
000050  122760  000000G 000000G                 CMPB    #CS.STA,C.STS(R0)       ; IS THIS A STATION TEMPLATE ?
000056  001371                                  BNE     10$                     ; IF NE, NO .. KEEP LOOKING
                                        ;
                                        ; ASSUME THIS IS THE RIGHT TEMPLATE
                                        ;
000060  011011                                  MOV     (R0),(R1)               ; CLOSE UP LIST
000062  010067  177712                          MOV     R0,TEMP                 ; SAVE TEMPLATE ADDRESS
000066  032760  000000G 000000G                 BIT     #CS.MOD,C.STS(R0)       ; WAS THIS TEMPLATE MODIFIED ?
000074  001004                                  BNE     15$                     ; IF NE, YES .. UPDATE THE RECORD
000076  012767  000000G 000000C                 MOV     #$BUFI,$OFDB+F.NRBD+2   ; ELSE, OUTPUT THE INPUT LINE
000104  000416                                  BR      30$                     ; AND LEAVE
000106  004767  000000G                 15$:    CALL    $MOVST                  ; MOVE FIRST FIELD INTO OUTPUT BUFFER
000112  010546                                  MOV     R5,-(SP)                ; SAVE R5
000114  005001                                  CLR     R1                      ; IGNORE BLANKS
000116  012702  000000'                         MOV     #STAKTB,R2              ; KEYWORD TABLE
000122  012705  000000'                         MOV     #STASTB,R5              ; STATE TABLE
000126  004767  000000G                         CALL    .TPARS                  ; PARSE THE REST OF THE LINE
000132  103002                                  BCC     20$                     ; NORMAL RETURN IF NO ERROR
000134  005267  000000G                         INC     $ERROR                  ; INDICATE SYNTAX ERROR
000140  012605                          20$:    MOV     (SP)+,R5                ; RESTORE R5
000142  000207                          30$:    RETURN
                                        ;
                                        ; ERROR CONDITIONS
                                        ;
000144  000167  000000G                 101$:   JMP     .CONER          ; CONSISTENCY ERROR
```

```
                        .SBTTL  MACRO CALLS AND LOCAL DATA
                        .SBTTL  MACRO DEFINITIONS
                ;
                ; LOCAL MACROS
                ;

                        .MACRO  PRINT   TEXT
                        MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                        CALL    $PRINT          ; PRINT MESSAGE
                        .ENDM   PRINT

                        .MACRO  ERROR$  TEXT
                ;
                ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
                ; MESSAGE STRING.
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ERROR MESSAGE
                        .ENDC
                        CALL    $CFERR                  ; PRINT ERROR MESSAGE
                        .ENDM   ERROR$

                        .MACRO  EPRINT  TEXT
                ;
                ; PRINT TEXT ON ERROR LUN
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                 ; GET ADDRESS OF ASCIZ STRING TO PRINT
                        .ENDC
                        CALL    $CFER1                  ; PRINT MESSAGE ON ERROR LUN
                        .ENDM   EPRINT
                ;
                ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
                ;
                        .MACRO  $$$PTN  ROU
                        .MACRO  'ROU    OFS,OCTL,OPT,?L
                        .IF NB <OPT>
                        MOV     OPT,R2
                        CALL    $PTOPT
                        BCS     L
                        .IFTF
                        MOV     OFS,R2
                        .IF NB <OCTL>
                        MOV     SP,R1
                        .IFF
                        CLR     R1
                        .ENDC ; IF NB <OCTL>
                        CALL    'ROU
                        .IFT  ; IF NB <OPT>
                L:
                        .ENDC ; IF NB <OPT>
                        .ENDM ; 'ROU
                        .ENDM ; $$$PTN

000000                  $$$PTN  $PTNUM
000000                  $$$PTN  $PTNMB
                ;
```

```
                                                .SBTTL  $UPUNT - UPDATE UNIT DEFINITION
                                        ;+
                                        ; *** - $UPUNT - UPDATE UNIT DEFINITION
                                        ;
                                        ; INPUT:
                                        ;       R3-R5   - TPARS REGISTERS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000000                                  $UPUNT::
000000  012700  000000G                         MOV     #$TMLST,R0      ; POINT AT TEMPLATE LIST
000004  010001                          10$:    MOV     R0,R1           ; SAVE ADDRESS OF CURRENT TEMPLATE
000006  011000                                  MOV     (R0),R0         ; GET NEXT TEMPLATE
000010  001435                                  BEQ     101$            ; IF EQ, END OF TEMPLATES
000012  122760  000000G 000000G                 CMPB    #CS.UNT,C.STS(R0) ; IS THIS A UNIT TEMPLATE ?
000020  001371                                  BNE     10$             ; IF NE, NO .. KEEP LOOKING
                                        ;
                                        ; ASSUME THIS IS THE RIGHT TEMPLATE
                                        ;
000022  011011                                  MOV     (R0),(R1)       ; REMOVE TEMPLATE FROM LIST
000024  010067  000000G                         MOV     R0,.TEMP        ; SAVE TEMPLATE
000030  005720                                  TST     (R0)+           ; POINT TO C.STS
000032  010067  000000G                         MOV     R0,.FLAGS       ; SAVE FOR OPTION TESTING
000036  004767  000000G                         CALL    $MOVST          ; MOVE FIRST FIELD INTO OUTPUT BUFFER
000042  010546                                  MOV     R5,-(SP)        ; SAVE R5
000044  005067  000000G                         CLR     .COMMA          ; SETUP FOR OPTIONS
000050  005001                                  CLR     R1              ; IGNORE BLANKS
000052  012702  000000'                         MOV     #UNTKTB,R2      ; KEYWORD TABLE
000056  012705  000000'                         MOV     #UNTSTB,R5      ; STATE TABLE
000062  004767  000000G                         CALL    .TPARS          ; PARSE THE REST OF THE LINE
000066  103002                                  BCC     20$             ; NORMAL RETURN IF NO ERROR
000070  005267  000000G                         INC     $ERROR          ; INDICATE SYNTAX ERROR
000074  012605                          20$:    MOV     (SP)+,R5        ; RESTORE R5
000076  004767  000000G                         CALL    $CLNUP          ; CALCULATE BUFFER LENGTH
000102  000207                          30$:    RETURN
                                        ;
                                        ; ERROR CONDITIONS
                                        ;
000104  000167  000000G                 101$:   JMP     .CONER          ; CONSISTENCY ERROR
```

Table of contents

```
     4-   57    MACRO CALLS AND LOCAL DATA
     4-  121    LOCAL DATA
     5-  145    $UPX2P - UPDATE X.25 LEVEL 2 PROTOCOL DEFINITION
     6-  181    SETLIN - SET LINE ID
```

```
                                        ;
                                                .MACRO  $PTR50  OFS,WC
                                                MOV     OFS,R2
                                                MOV     WC,R1
                                                CALL    $PTR50
                                                .ENDM

```

```
                                                                     L 12
UPDX3P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE               REFERENCES

C.CT       000012            #4-132     5-174
C.DFP      000004            #4-128     5-170
C.DW       000010            #4-130     5-172
C.KM       000016            #4-136     5-178
C.KT       000013            #4-133     5-175
C.LEN    = 000021            #4-139
C.MW       000011            #4-131     5-173
C.MXP      000006            #4-129     5-171
C.RM       000017            #4-137     5-179
C.RT       000014            #4-134     5-176
C.SM       000020            #4-138     5-180
C.ST       000015            #4-135     5-177
F.NRBD   = ******    GX      *5-160    *5-183    *5-185
X3PDF      000000 R          #4-142     4-143     5-166
X3PLN    = 000010            #4-143     5-167
$BUFI    = ******    GX       5-185
$MVASC   = ******    GX       5-168
$OBUF    = ******    GX      *5-169     5-181
$OFDB    = ******    GX      *5-160    *5-183    *5-185
$PTNMB   = ******    GX       5-172     5-173     5-174     5-175     5-176     5-177     5-178     5-179     5-180
$PTNUM   = ******    GX       5-170     5-171
$UPX3P     000010 RG         #5-159
$WDATA   = ******    GX       5-184
.BUFF    = ******    GX       5-160     5-165     5-182
.COMMA   = ******    GX      *5-164
.FLAGS   = ******    GX      *5-163
.TEMP    = ******    GX      *5-161
```

```
                                                                     M 12
UPDX3P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59      PAGE 2
```

```
 000372  010167  001376'                   MOV     R1,CURQST              ;; Save address of CCB
 000376  026127  000010  002026            CMP     C.FNC(R1),#FC.MLD+FS.RLB;; Is this a microcode load request?
 000404  001354                            BNE     15$                    ;; Branch if not

                                           .IF     DF      N$$UMR         ;; If system has UMRs ...

 000406                                    RETURN                         ;; Return to user state at "50$"

 000410  016704  001364'          40$:     MOV     LDBPTR,R4              ;; Get address of CCB/LDB pair
 000414                                    CALL    @LDBRT                 ;; Return CCB and LDB to pool
 000420  005067  001364'                   CLR     LDBPTR                 ;; LDB no longer available
 000424                                    RETURN                         ;; Return to user state at "50$"

                                           .IFF                           ;; If system has no UMRs ...

                                  40$:     RETURN                         ;; Return to user state at "50$"

                                           .ENDC                          ;; End UMR conditional


                                  ;
                                  ;        If a load request has been successfully dequeued,  look up all
                                  ;        necessary information for dispatches to the device driver.  This
                                  ;        lookup is done once to save a search each time a dispatch is needed.
                                  ;

 000426  016704  001376'          50$:     MOV     CURQST,R4              ; Get address of current request
 000432  001461                            BEQ     90$                    ; Branch if none
 000434  116403  000006                    MOVB    C.LIN(R4),R3           ; Get system line number

                                           .IF     DF      N$$UMR         ; If system has UMRs ...

 000440  016705  001364'                   MOV     LDBPTR,R5              ; Get address of CCB/LDB
 000444  010365  000006                    MOV     R3,C.LIN(R5)           ; Stuff in system line number

                                           .ENDC                          ; End UMR conditional

 000450  006303                            ASL     R3                     ; Form a word index with SLN
 000452  067703  177354                    ADD     @SLTMA,R3              ; Get pointer to SLT vector address
 000456  011303                            MOV     (R3),R3                ; Get address of SLT vector

 000460  116367  000002  001400'           MOVB    L.DDM(R3),PDVIDX       ; Get driver's PDV index
 000466  016367  000004  001404'           MOV     L.DDS(R3),DDMTBL       ; Save driver's line table address
 000474  116367  000012  001402'           MOVB    L.CTL(R3),CTLNUM       ; Save device controller number

                                           .IF     DF      R$$MPL         ; If system is RSX-11M+ ...

                                  ;
                                  ;        On RSX-11M+ systems with multiple CPUs,  we must make sure that
                                  ;        MLD is running on the same processor that the device to be loaded
                                  ;        is on.  Get the UNIBUS run mask of the device from the KRB data
                                  ;        structure pointed to by the SLT entry for the line.
                                  ;

                                           TSTB    @NCPU                  ; More than one CPU in configuration?
                                           BEQ     60$                    ; Branch if not
                                           MOV     L.KRBA(R3),R2          ; Get controller request block address
                                           RMAF$S                         ; Remove previous CPU/UNIBUS affinity
```

```
                        .SBTTL  SLOAD   - Load Control for RSX-11S Systems
                ;+
                ;                               *** SLOAD ***
                ;
                ; Function:     To control microcode load sequences on RSX-11S systems.
                ;               This routine segments a microcode partition into 4K word pieces
                ;               for dispatch to the device driver and maintains the load seq-
                ;               uence counter(s) and status.
                ;
                ; Call Format:  CALL    SLOAD
                ;
                ; Inputs:       PCBADR  = address of partition's PCB
                ;               R4 = address of "load request" CCB
                ;
                ; Outputs:      Load sequence ended
                ;               R3 = load halt status
                ;               R4 = address of "halt load" CCB
                ;
                ; Registers:    R0 through R5 are destroyed
                ;
                ;-

                SLOAD:  MOV     #NBUFS,R0               ; Set up a loop counter
                        MOV     #BLOCK,R2               ; Get address of load control words
                        ADD     #C.BUF1,R4              ; Point at first CCB buffer descriptor

                10$:    MOV     (R2)+,(R4)+             ; Copy bias of next partition segment
                        MOV     #140000,(R4)+           ; Starting address is virtual 0 (APR6)
                        MOV     #177,R3                 ; Assume segment size =177*32 =4k words
                        CMP     R3,(R2)                 ; Is that greater than what's left?
                        BGT     20$                     ; Branch if so and do some adjusting
                        SUB     R3,(R2)                 ; Compute # 32-word blocks left to load
                        BR      30$                     ; Branch to common code

                20$:    MOV     (R2),R3                 ; Load segment size = partition remnant
                        CLR     (R2)                    ; No data left to load

                30$:    ADD     R3,-(R2)                ; Compute bias of next load segment
                        .REPT   6                       ; Compute number of bytes to load
                        ASL     R3                      ;   from number of 32-word blocks
                        .ENDR                           ;     in current load segment
                        MOV     R3,(R4)+                ; Move segment size in bytes to CCB
                        CLR     (R4)+                   ; Clear CCB flag word
                        TST     BLOCK+2                 ; Has end of partition been reached?
                        BEQ     40$                     ; Branch if so
                        SOB     R0,10$                  ; Loop around if more to load

                40$:    MOV     #CF.LB,-(R4)            ; Mark last descriptor as last buffer
                        MOV     CURQST,R4               ; Retrieve starting address of CCB
                        CALL    DSPDRV                  ; Call driver to load microcode
                        BCS     50$                     ; Branch if driver halted the load
                        TST     BLOCK+2                 ; Has the entire partition been loaded?
                        BNE     SLOAD                   ; Branch if not
                        CLR     R3                      ; Load halt status = end of data

                50$:    RETURN                          ; Exit
```

```
                                                                   L 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 8
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL   VALUE             REFERENCES

ZF.MUX   = 000040          #6-84
ZF.PSE   = 002000          #6-84
ZF.SLI   = 010000          #6-84
ZF.TIM   = 000200          #6-84
ZF.X3P   = 000000          #6-84
ZS.ASN   = 100000          #6-84
ZS.BSY   = 140000          #6-84
Z.AVL      000014          #6-84
Z.DAT      000016          #6-84
Z.DSP      000000          #6-84      6-84
Z.FLG      000010          #6-84
Z.LEN    = 000016          #6-84
Z.LLN      000006          #6-84
Z.MAP      000020          #6-84
Z.NAM      000004          #6-84     11-604
Z.PCB      000012          #6-84
Z.SCH      000007          #6-84
$BLXIO   = ******   GX      8-230
$CBTA    = ******   GX      9-296
$CCBRT   = ******   GX      8-238
$DIV     = ******   GX      9-290
$EDMSG   = ******   GX      11-609    13-721
$LDBGT   = ******   GX      8-239
$LDBRT   = ******   GX      8-240
$NTUIC   = ******   GX      8-234
$PARHD   = ******   GX      8-235
$PDSPL   = ******   GX      8-241
$PDVTA   = ******   GX      8-242
$QRMVF   = ******   GX      8-236
$RDBSZ   = ******   GX      8-243
$SLTMA   = ******   GX      8-244
$STDDM   = ******   GX      8-245
$TKTCB   = ******   GX      8-237
$$$ARG   = 000002          #11-617    11-617   #11-617    11-617   #11-617    11-617   #11-617    11-617   #11-617
                            11-617    11-617   #11-617    11-617    11-617    11-617    11-617    11-617    11-617
                            11-617    11-617    11-617    11-617   #11-617    11-617    11-617    11-617    11-617
                            11-617    11-617    11-617    11-617    11-617    11-617   #11-617    11-617    11-617
                            11-617    11-617    11-617    11-617    11-617    11-617    11-617    11-617
$$$GLB   = ******           8-202     8-203
$$$T1    = 000067          #14-765    14-765    14-765    14-765   #15-794    15-794    15-794    15-794
.CLOSE   = ******   G       12-651    12-651    15-808    15-808
.FINIT   = ******   G       9-283     9-283
.FSRCB   = ******   G       8-221
.OPEN    = ******   G       14-765    14-765
.PPASC   = ******   GX      9-304
.READ    = ******   G       15-794    15-794
.WAIT    = ******   G       15-796    15-796
...PC1   = 000144 R        #8-166     8-168     8-169     8-169     8-169     8-169     8-170     8-170
...PC2   = 000304 R        #8-168     8-168    #8-169     8-169    #8-169     8-169    #8-169     8-169    #8-169
                            8-169    #8-170     8-170    #8-170     8-170
```

```
                                                                   M 15
MLD          CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 9
```

```
                                                MOV     R2,DSDS+4               ; Move length to dataset descriptor

                                                .IFTF                           ; For any operating system ..

     000020  005067  000156'                    CLR     CURQST                  ; No load request outstanding

                                                .IF     DF      N$$UMR          ; If system has UMRs ...

     000024  005067  000144'                    CLR     LDBPTR                  ; No large data buffer yet

                                                .ENDC                           ; End UMR conditional

     000030                                     SPRA$S  #PWRAST                 ; Specify AST for powerfail recovery
     000042                                     SREA$S  #ABOAST                 ; Specify AST for requested aborts

                                                .IF     DF      N$$UMR          ; If system has UMRs ...



                                        ;       In order to guarantee that devices which DMA in their own microcode
                                        ;       can access the microcode,  allocate a large data buffer before
                                        ;       continuing.  LDBs are always UMR-mapped if the system has UMRs.


     000054                             5$:     CALL    $SWSTK,7$               ;; Switch to system state
     000060                                     CALL    @LDBGT                  ;; Allocate a UMR-mapped buffer
     000064  103413                             BCS     6$                      ;; Branch if failure
     000066  010467  000144'                    MOV     R4,LDBPTR               ;; Save address of CCB/LDB
     000072  016467  000014  000146'            MOV     C.BUF1(R4),LDBIAS       ;; Save LBD's mapping bias
     000100  016467  000016  000150'            MOV     C.BUF1+2(R4),LDBADR     ;; Save LDB's starting address
     000106  012764  001026  000010             MOV     #FC.MLD+FS.WLB,C.FNC(R4);; LDB used only for writes

     000114                             6$:     RETURN                          ;; Return to user state at "7$"
     000116  005767  000144'            7$:     TST     LDBPTR                  ;  Was an LDB allocated?
     000122  001007                             BNE     10$                     ;  Branch if so
     000124                                     DIR$    #SETTMR                 ;  Set a timer
     000132                                     DIR$    #WAIT                   ;  Wait for timer to expire
     000140  000745                             BR      5$                      ;  Attempt LDB allocation again

                                                .ENDC                           ;  End UMR conditional



                                        ;       Enter system state to dequeue a CCB from our TCB receive list.
                                        ;       Each CCB is a load request queued to MLD by AUX on behalf of
                                        ;       a DECnet device driver.


     000142                             10$:    CALL    $SWSTK,50$              ;; Switch to system state
     000146  016704  000156'            15$:    MOV     CURQST,R4               ;; Get current request CCB
     000152  001404                             BEQ     20$                     ;; Branch if none there
     000154                                     CALL    @CCBRT                  ;; Return CCB to its pool
     000160  005067  000156'                    CLR     CURQST                  ;; Request no longer outstanding

     000164  017700  000602'            20$:    MOV     @TKTCB,R0               ;; Get address of TCB
     000170  062700  000012                     ADD     #T.RCVL,R0              ;; Advance to TCB receive listhead

     000174                             30$:    CALL    @QRMVF                  ;; Dequeue a load request CCB
     000200  103407                             BCS     40$                     ;; Branch if no CCB(s) queued
```

```
    193                                                 .SBTTL  SETSTA - SET STATE
    194                                         ;+
    195                                         ; ** SETSTA - SET STATE
    196                                         ;
    197                                         ; INPUT:        CURRENT TEMPLATE.
    198                                         ;
    199                                         ; OUTPUT:       VALUE LOADED INTO OUTPUT BUFFER.
    200                                         ;
    201                                         ;-
    202
    203 000402  016701  000000G         SETSTA: MOV     .TEMP,R1                ; POINT TO TEMPLATE
    204 000406  010146                          MOV     R1,-(SP)                ; SAVE TEMPLATE ADDRESS
    205 000410  005002                          CLR     R2                      ; ASSUME STATE = ON
    206 000412  105761  000000G                 TSTB    P$VSTA(R1)              ; CHECK STATE
    207 000416  001001                          BNE     10$                     ; IF NE, ON
    208 000420  122222                          CMPB    (R2)+,(R2)+             ; ELSE STATE = OFF
    209 000422  012767  000010' 000000G 10$:    MOV     #ONSTR,.TEMP            ; POINT TO START OF STRINGS
    210 000430  012701  000001                  MOV     #1,R1                   ; ONE RAD50 WORD
    211 000434  004767  000000G                 CALL    $PTR50                  ; OUTPUT RAD50
    212 000440  012667  000000G                 MOV     (SP)+,.TEMP             ; RESTORE TEMPLATE ADDRESS
    213 000444  000207                  20$:    RETURN
    214
    215         000001                          .END
```

```
UPDRDT     CREATED BY  MACRO  ON 29-JUN-85 AT 05:55      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

EPRINT            #4-79
ERROR$            #4-68
PRINT             #4-63
$PTNMB            #4-112
$PTNUM            #4-111
$PTR50            #4-116      6-173      6-174
$$$PTN            #4-91       4-111      4-112
```

```
**FILE**ID**UPDREM
```

```
                                                                    M  3
UPDREM      CREATED BY  MACRO  ON 29-JUN-85 AT 05:55        PAGE  1
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL   VALUE              REFERENCES

BUFF       000033 R         #4-81       6-164       6-177       6-192
CS.FIL   = ******   GX       10-309
CS.REM   = ******   GX       6-170
CS.SER   = ******   GX       10-299
C.ADD      000010           #4-91       8-244       9-270       10-301
C.LEN    = 000012           #4-92
C.NAM      000004           #4-90       7-220       7-223
C.STS    = ******   GX       6-170      10-299      10-309
F.NRBD   = ******   GX      *6-164     *6-193      *6-199
LA       = 000074           #4-74
RA       = 000076           #4-75
REMDEF     000002 R         #4-77       4-78        6-185
REMDF      000023 R         #4-79       4-80        6-178
REMKTB     000000 RG        #5-120      6-183
REMLEN   = 000020           #4-78       6-184
REMLN    = 000010           #4-80       6-179
REMSTB     000000 RG        #5-120      6-186
SETADD     000444 R         #9-267
SETARE     000404 R         #8-241
SETNAM     000342 R         #7-217
S$ADD      000[illegible]           #4-102      10-301
S$CIR      000[illegible]2           #4-105
S$DEV      000[illegible]           #4-106
S$DPA      00005[illegible]           #4-108
S$DPC      000054           #4-109
S$FLG      000004           #4-101
S$HAD      000012           #4-104
S$HST      000010           #4-103
S$LEN    = 000056           #4-110
S$PSS      000034           #4-107
TEMP       000000 R         #4-76      *6-176       7-219       7-222       8-243       9-269       10-294
UPDLL      000476 R          6-195     #10-291
$ALPHA   = 000022           #5-120
$ANY     = 000020           #5-120
$BLANK   = 000006           #5-120
$BUFI    = ******   GX       6-199
$CNVD    = ******   GX       8-249      9-272
$C5TA    = ******   GX       7-221      7-224
$DIGIT   = 000024           #5-120
$DNUMB   = 000014           #5-120
$EOS     = 000012           #5-120
$ERROR   = ******   GX      *6-189     *6-197
$EXIT    = 000000           #5-120
$FAIL    = 177777           #5-120
$GPRM    = ******            5-120
$LAMDA   = 000000           #5-120
$MVASC   = ******   GX       6-180
$NUMBR   = 000002           #5-120
$OBUF    = ******   GX      *6-181      6-191       7-218      *7-225       8-242      *8-250       9-268      *9-273
$OFDB    = ******   GX      *6-164     *6-193      *6-199
$RAD50   = 000016           #5-120
$RCNLY   = ******            5-120      5-120       5-120
```

```
                                                                    N  3
UPDREM      CREATED BY  MACRO  ON 29-JUN-85 AT 05:55        PAGE  2
```

```
UPDROU      CREATED BY  MACRO  ON 29-JUN-85 AT 05:56      PAGE 2
MACRO CROSS REFERENCE                                     CREF   04.00
MACRO NAME         REFERENCES

DBGTP$             #5-100
ISTAT$             #4-67        5-100
MTRAN$             #5-100
PRINT              #4-58
STATE$             #4-67        5-103     #5-106    #5-109    #5-112    #5-115    #5-118    #5-121    #5-124    #5-127
                   #5-130      #5-133     #5-136    #5-139    #5-142    #5-145    #5-147    #5-150    #5-155    #5-158
TRAN$              #4-67       #5-104     #5-107    #5-110    #5-113    #5-116    #5-119    #5-122    #5-125    #5-128
                   #5-131      #5-134     #5-137    #5-140    #5-143    #5-146    #5-148    #5-151    #5-156
```

```
**FILE**ID**UPDSER
```

```
UPDSER     CREATED BY  MACRO  ON 29-JUN-85 AT 05:56     PAGE 2
MACRO CROSS REFERENCE                                    CRE:    04.00

MACRO NAME      REFERENCES

RESRG           #4-49        7-219        8-245
SAVRG           #4-49        7-207        8-233
```

```
**FILE**ID**UPDSLT
```

```
                                                .SBTTL  SETUNT - CHECK THE UNIT NUMBER
                                        ;+
                                        ; *** - SETUNT - CHECK THE UNIT NUMBER
                                        ;
                                        ; INPUT:
                                        ;               .PNUMB = UNIT NUMBER
                                        ;               .PNUMH = UNIT NUMBER (HIGH ORDER)
                                        ;
                                        ; OUTPUT:
                                        ;               C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000566                                  SETUNT:
000566  004767  000000G                         CALL    $MOVPT          ; MOVE THE UNIT NUMBER INTO THE OUTPUT BUFFER
000572  016700  000000G                         MOV     .TEMP,R0        ; GET START OF TEMPLATE
000576  126760  000000G 000011                  CMPB    .PNUMB,C.UNT(R0) ; IS THIS STILL THE RIGHT TEMPLATE ?
000604  001402                                  BEQ     10$             ; IF EQ, YES .. OKAY
000606  062716  000002                          ADD     #2,(SP)         ; ELSE, REJECT THE TRANSITION
000612  000207                          10$:    RETURN

                                                .SBTTL  SETCTT - SET OPTIONAL COUNTER TIMER VALUE
                                        ;+
                                        ; *** - SETUNT - SET THE OPTIONAL COUNTER TIMER
                                        ;
                                        ; INPUT:
                                        ;               .PNUMB = UNIT NUMBER
                                        ;               .PNUMH = UNIT NUMBER (HIGH ORDER)
                                        ;
                                        ; OUTPUT:
                                        ;               C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000614                                  SETCTT: $PTNUM  #S$LCTT,,#SL$CTM
000640  000207                                  RETURN

                                                .SBTTL  SETHTM - SET OPTIONAL HELLO TIMER VALUE
                                                .SBTTL  SETRPR - SET OPTIONAL ROUTER PRIORITY
                                                .SBTTL  SETBRA - SET BROADCAST ROUTER ADJACENCIES
                                        ;+
                                        ; *** - SETUNT - SET THE OPTIONAL HELLO TIMER
                                        ;
                                        ; INPUT:
                                        ;               .PNUMB = UNIT NUMBER
                                        ;               .PNUMH = UNIT NUMBER (HIGH ORDER)
                                        ;
                                        ; OUTPUT:
                                        ;               C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-

000642                                  SETHTM: $PTNUM  #S$LHTM,,#SL$HTM
000666  000207                                  RETURN

000670  016700  000000G                 SETRPR: MOV     .TEMP,R0
000674  032760  000000G 000000G                 BIT     #LF.BRO,S$LFLG(R0)
```

```
                                         .SBTTL  SETADD - SET UP STATION ADDRESS
                                 ;+
                                 ; *** - SETADD - SETUP STATION ADDRESS
                                 ;
                                 ; THIS ACTION ROUTINE SETS UP THE STATION ADDRESS
                                 ;
                                 ; INPUT:
                                 ;        C.ADD = STATION ADDRESS
                                 ;        $OBUF = CURRENT POSITION IN OUTPUT BUFFER
                                 ;
                                 ; OUTPUT:
                                 ;        C.ADD STORED IN OUTPUT BUFFER
                                 ;        $OBUF IS UPDATED
                                 ;
                                 ;-

000150                           SETADD:
000150  016702  177624                   MOV     TEMP,R2             ; GET TEMPLATE ADDRESS
000154  005001                           CLR     R1                  ; GET THE STATION ADDRESS
000156  156201  000005                   BISB    C.ADD(R2),R1        ; ..WITHOUT SIGN EXTEND
000162  016700  000000G                  MOV     $OBUF,R0            ; GET CURRENT POSTITION IN OUTPUT BUFFER
000166  004767  000000G                  CALL    $CNVD               ; CONVERT NUMBER TO OCTAL
000172  010067  000000G                  MOV     R0,$OBUF            ; UPDAT THE POINTER INTO OUTPUT BUFFER
000176  000207                           RETURN
```

```
; PUT RAD50 TEMPLATE FIELD TO BUFFER
;
              .MACRO  $PTR50  OFS,WC,OPT,?L
              .IF NB <OPT>
              MOV     OPT,R2
              CALL    $PTOPT
              BCS     L
              .IFTF
              MOV     OFS,R2
              MOV     WC,R1
              CALL    $PTR50
              .IFT
L:
              .ENDC
              .ENDM

```

```
                                                .SBTTL  SETUNT - CHECK UNIT NUMBER
                                        ;+
                                        ; *** - SETUNT - CHECK UNIT NUMBER
                                        ;
                                        ; INPUT:
                                        ;       .PNUMB = UNIT NUMBER
                                        ;       .PNUMH = (HIGH ORDER)
                                        ;       .TEMP = TEMPLATE ADDRESS
                                        ;
                                        ; OUTPUT:
                                        ;       C-BIT = SUCCESS/FAILURE
                                        ;
                                        ;-
 000110                                 SETUNT:
 000110  004767  000000G                        CALL    $MOVPT          ; MOVE THE UNIT NUMBER INTO THE OUTPUT BUFFER
 000114  016700  000000G                        MOV     .TEMP,R0                ; GET TEMPLATE ADDRESS
 000120  026760  000000G 000004                 CMP     .PNUMB,C.UNT(R0) ; IS T 'S THE RIGHT UNIT TEMPLATE ?
 000126  001402                                 BEQ     10$             ; IF EQ. YES .. OKAY
 000130  062716  000002                         ADD     #2,(SP)         ; ELSE, REJECT THE TRANSITION
 000134  000207                         10$:    RETURN
```

```
                .TITLE  UPDX2P - UPDATE X.25 LEVEL 2 CHARACTERISTICS
                .IDENT  /V05.00/
                .ENABL  LC

;
; COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       UPDATE X.25 LEVEL 2 CHARACTERISTICS IN CETAB
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-AUG-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 3.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                        .SBTTL  LOCAL DATA

     000000       011      130      062  X29DF:  .ASCII  <11>/X29$DF/<11>
     000003       071      044      104
     000006       106      011
                000010                  X29LN=.-X29DF
                                        .LIST   BEX
                                        .EVEN
```

M 12

```
UPDX3P     CREATED BY  MACRO  ON 29-JUN-85 AT 05:59     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

EPRINT           #4-78
ERROR$           #4-67
PRINT            #4-62
$PTNMB           #4-111      5-172      5-173      5-174      5-175      5-176      5-177      5-178      5-179      5-180
$PTNUM           #4-110      5-170      5-171
$PTR50           #4-115
$$$PTN           #4-90       4-110      4-111
```

N 12

```
                                        STAF$S  K.URM(R2)               ; Set new CPU/UNIBUS affinity

                                        .ENDC                           ; End RSX-11M+ conditional


                                ;       Now locate the microcode in the system.  If the microcode has been
                                ;       successfully located,  perform the load cycle.

000502  062704  000014          60$:    ADD     #C.BUF1,R4              ; Point at microcode ASCII name length
000506  012405                          MOV     (R4)+,R5                ; Copy it and point at start of name

                                        .IFT                            ; If system is RSX-11M ...

000510                                  CALL    LOCFIL                  ; Locate microcode file

                                        .IFF                            ; If system is RSX-11S ...

                                        CALL    LOCPAR                  ; Locate microcode partition

                                        .IFTF                           ; For any operating system ...

000514  016704  001376'                 MOV     CURQST,R4               ; Recover CCB address
000520  103003                          BCC     70$                     ; Branch if microcode was found
000522  012703  001000                  MOV     #CS.FTL,R3              ; Else, get "fatal error" return
000526  000407                          BR      80$                     ; Branch to common code

000530  012764  001026  000010  70$:    MOV     #FC.MLD+FS.WLB,C.FNC(R4); Change CCB (sub)function

                                        .IFT                            ; If system is RSX-11M ...

000536                                  CALL    MLOAD                   ; Load microcode from disk

                                        .IFF                            ; If system is RSX-11S ...

                                        CALL    SLOAD                   ; Load microcode from partition

                                        .IFTF                           ; For any operating system ...

000542  005703                          TST     R3                      ; Did the driver stop the load?
000544  100672                          BMI     10$                     ; Branch if so, save a dispatch

000546  016704  001376'         80$:    MOV     CURQST,R4               ; Get address of current CCB
000552  012764  000026  000010          MOV     #FC.MLD+FS.HLT,C.FNC(R4); Set CCB function to "load halted"
000560  052703  100000                  BIS     #CS.ERR,R3              ; "OR" general error flag into status
000564  010364  000012                  MOV     R3,C.STS(R4)            ; Pass load status
000570                                  CALL    DSPDRV                  ; Dispatch to device driver
000574  000656                          BR      10$                     ; Look for another load request

000576                          90$:    EXIT$S                          ; Program exit
```

```
    988                                         .ENDC                                   ; End RSX-11S conditional
    989         000036'                         .END    MLD                             ; End of source file
```

```
MACRO CROSS REFERENCE                                  CREF    04.00

MACRO NAME      REFERENCES

ALUN$S          #6-78        9-269
BDOFF$          #8-221
CALL             9-290       9-296       9-304       9-329       9-330       9-352       9-355       9-361       9-372       9-434
                 9-451       9-466       10-499      10-529      10-574      11-609      12-655      12-661      12-670      12-674
                 12-680      12-682      13-721      13-724      14-765      15-794      15-805
CALLR            10-547
CBYTE$          #8-167       8-167       8-167      #8-168       8-168      #8-169       8-169      #8-170       8-170       8-170
CCBDF$          #6-79        6-82
CLOSE$          #6-64        12-651      15-808
CMOV$B          #14-765     #14-765      14-765      14-765     #14-765      14-765     #15-794      15-794     #15-796      15-796
CMOV$W          #14-765      14-765      14-765      14-765     #14-765      14-765      14-765     #15-794      15-794      15-794
                 15-794      15-794     #15-796      15-796
CMOV$2          #15-794      15-794
WORD$           #8-167       8-167       8-167       8-167      #8-168       8-168       8-168      #8-169       8-169       8-169
                 8-169       8-169       8-169      #8-170       8-170       8-170       8-170
CBDF$           #6-65        6-68
EF$L            #8-221
DIR$            #6-89       #9-269       9-269      #9-318       9-318      #9-319       9-319       9-340       9-341      #9-469
                 9-469      #11-617      11-617     #12-687      12-687
ERR$            #12-651      12-651     #14-765      14-765     #15-794      15-794     #15-796      15-796     #15-808      15-808
EVLDF$          #6-79        6-83
EXIT$S          #6-78        9-469       12-687
FDAT$A          #6-65        8-167
FDBDF$          #6-64        8-166
FDBK$A          #6-65        8-169
FDBK$R          #15-794      15-794
FDBSZ$          #8-166       8-166
FDOFF$          #8-167       8-167      #8-168       8-168      #8-169       8-169      #8-170       8-170
FDOP$A          #6-65        8-170
FDOP$R          #14-765      14-765
FDRC$A          #6-65        8-168
FDRC$R          #14-765      14-765
FINIT$          #6-65        9-283
FSRSZ$          #6-64        8-221
LDFDB$          #12-651      12-651     #14-765      14-765     #14-765      14-765     #15-794      15-794     #15-796      15-796
                #15-808      15-808
MOV$            #9-269       9-269       9-269       9-269      #9-318       9-318      #9-319       9-319      #11-617      11-617
                 11-617      11-617      11-617     #11-617      11-617      11-617     #11-617      11-617      11-617
MRKT$           #6-89        8-202
MVB$            #11-617      11-617
OFF$            #8-202      #8-203
OPEN$           #14-765      14-765
OPEN$R          #6-64        14-765
PDVDF$          #6-79        6-84
QDPB$S          #11-617      11-617
QIOW$S          #6-78        11-617
READ$           #6-64        15-794
RETURN           9-337       9-369       9-374       10-562      10-577      11-618      12-685      13-728      14-770      15-809
RVP$            #11-617      11-617      11-617      11-617
SLTDF$          #6-79        6-85
SPRA$S          #6-78        9-318
SREA$S          #6-78        9-319
```

```
 000202  010167  000156'                      MOV     R1,CURQST              ;; Save address of CCB
 000206  026127  000010  002026               CMP     C.FNC(R1),#FC.MLD+FS.RLB;; Is this a microcode load request?
 000214  001354                               BNE     15$                    ;; Branch if not

                                              .IF     DF      N$$UMR         ;; If system has UMRs ...

 000216                                       RETURN                         ;; Return to user state at "50$"

 000220  016704  000144'             40$:     MOV     LDBPTR,R4              ;; Get address of CCB/LDB pair
 000224                                       CALL    @LDBRT                 ;; Return CCB and LDB to pool
 000230  005067  000144'                      CLR     LDBPTR                 ;; LDB no longer available
 000234                                       RETURN                         ;; Return to user state at "50$"

                                              .IFF                           ;; If system has no UMRs ...

                                     40$:     RETURN                         ;; Return to user state at "50$"

                                              .ENDC                          ;; End UMR conditional


                                     ;        If a load request has been successfully dequeued, look up all
                                     ;        necessary information for dispatches to the device driver.  This
                                     ;        lookup is done once to save a search each time a dispatch is needed.


 000236  016704  000156'             50$:     MOV     CURQST,R4              ; Get address of current request
 000242  001461                               BEQ     90$                    ; Branch if none
 000244  116403  000006                       MOVB    C.LIN(R4),R3           ; Get system line number

                                              .IF     DF      N$$UMR         ; If system has UMRs ...

 000250  016705  000144'                      MOV     LDBPTR,R5              ; Get address of CCB/LDB
 000254  010365  000006                       MOV     R3,C.LIN(R5)           ; Stuff in system line number

                                              .ENDC                          ; End UMR conditional

 000260  006303                               ASL     R3                     ; Form a word index with SLN
 000262  067703  000620'                      ADD     @SLTMA,R3              ; Get pointer to SLT vector address
 000266  011303                               MOV     (R3),R3                ; Get address of SLT vector

 000270  116367  000002  000160'              MOVB    L.DDM(R3),PDVIDX       ; Get driver's PDV index
 000276  016367  000004  000164'              MOV     L.DDS(R3),DDMTBL       ; Save driver's line table address
 000304  116367  000012  000162'              MOVB    L.CTL(R3),CTLNUM       ; Save device controller number

                                              .IF     DF      R$$MPL         ; If system is RSX-11M+ ...


                                     ;        On RSX-11M+ systems with multiple CPUs,  we must make sure that
                                     ;        MLD is running on the same processor that the device to be loaded
                                     ;        is on.  Get the UNIBUS run mask of the device from the KRB data
                                     ;        structure pointed to by the SLT entry for the line.


                                              TSTB    @NCPU                  ; More than one CPU in configuration?
                                              BEQ     60$                    ; Branch if not
                                              MOV     L.KRBA(R3),R2          ; Get controller request block address
                                              RMAF$S                         ; Remove previous CPU/UNIBUS affinity
```

```
                                                                                              N  1
UPDPVC - UPDATE PERMANENT VIRTU MACRO V05.03b Saturday 29-Jun-85 05:55  Page 7-1
Symbol table

CS.FLG= ****** GX       PV$WSZ= ****** GX       P$VPSZ= ****** GX       $OBUF = ****** GX       $UPPVL  000014RG
F.NRBD= ****** GX       P$VCT = ****** GX       P$VSTA= ****** GX       $OFDB = ****** GX       $WDATA= ****** GX
ONSTR   000010R         P$VFLG= ****** GX       P$VWSZ= ****** GX       $PTNMB= ****** GX       .BUFF = ****** GX
PVCDF   000000R         P$VLCN= ****** GX       SETSTA  000402R         $PTNUM= ****** GX       .COMMA= ****** GX
PVCLN = 000010          P$VNAM= ****** GX       $BUFI = ****** GX       $PTOPT= ****** GX       .FLAGS= ****** GX
PV$OWN= ****** GX       P$VOWN= ****** GX       $MVASC= ****** GX       $PTR50= ****** GX       .TEMP = ****** GX
PV$PSZ= ****** GX

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000446    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8838 Words  ( 35 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:06.44
SY:UPDPVC.V2,[132,134]UPDPVC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDPVC
```

```
                                                                              N 2
**FILE**ID**UPDREM
```

```
UU      UU   PPPPPPPP     DDDDDDDD     RRRRRRRR     EEEEEEEEEE   MM      MM
UU      UU   PPPPPPPP     DDDDDDDD     RRRRRRRR     EEEEEEEEEE   MM      MM
UU      UU   PP      PP   DD      DD   RR      RR   EE           MMMM  MMMM
UU      UU   PP      PP   DD      DD   RR      RR   EE           MMMM  MMMM
UU      UU   PP      PP   DD      DD   RR      RR   EE           MM  MM  MM
UU      UU   PP      PP   DD      DD   RR      RR   EE           MM  MM  MM
UU      UU   PPPPPPPP     DD      DD   RRRRRRRR     EEEEEEEE     MM      MM
UU      UU   PPPPPPPP     DD      DD   RRRRRRRR     EEEEEEEE     MM      MM
UU      UU   PP           DD      DD   RR  RR       EE           MM      MM
UU      UU   PP           DD      DD   RR  RR       EE           MM      MM
UU      UU   PP           DD      DD   RR    RR     EE           MM      MM      ....
UU      UU   PP           DD      DD   RR    RR     EE           MM      MM      ....
UUUUUUUUUU   PP           DDDDDDDD     RR      RR   EEEEEEEEEE   MM      MM      ....
UUUUUUUUUU   PP           DDDDDDDD     RR      RR   EEEEEEEEEE   MM      MM      ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

N 3

```
UPDREM      CREATED BY  MACRO  ON 29-JUN-85 AT 05:55      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

$SAVRG  = ******  GX         10-292
$STRNG  = 000004            #5-120
$SUBXP  = 000010            #5-120
$TMLST  = ******  GX          6-166       10-293
$UPFIL  = ******  GX         10-311
$UPREM    000154 RG         #6-163
$UPSER  = ******  GX         10-306
$WDATA  = ******  GX          6-194
$$$FLG  = 177777            #5-120
$$$KEY  = 177777            #5-120
.CONER  = ******  GX         10-314
.TPARS  = ******  GX          6-187
```

```
                                                                              N  4
**FILE**ID**UPDSER




UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   EEEEEEEEEE   RRRRRRRR
UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   EEEEEEEEEE   RRRRRRRR
UU      UU   PP      PP   DD      DD   SS           EE           RR      RR
UU      UU   PP      PP   DD      DD   SS           EE           RR      RR
UU      UU   PP      PP   DD      DD   SS           EE           RR      RR
UU      UU   PP      PP   DD      DD   SS           EE           RR      RR
UU      UU   PPPPPPPP     DD      DD     SSSSSS     EEEEEEEE     RRRRRRRR
UU      UU   PPPPPPPP     DD      DD     SSSSSS     EEEEEEEE     RRRRRRRR
UU      UU   PP           DD      DD           SS   EE           RR  RR
UU      UU   PP           DD      DD           SS   EE           RR  RR
UU      UU   PP           DD      DD           SS   EE           RR    RR          ....
UU      UU   PP           DD      DD           SS   EE           RR    RR          ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS     EEEEEEEEEE   RR      RR        ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS     EEEEEEEEEE   RR      RR        ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                N 5
**FILE**ID**UPDSLT






UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   LL           TTTTTTTTTT
UU      UU   PPPPPPPP     DDDDDDDD       SSSSSSSS   LL           TTTTTTTTTT
UU      UU   PP      PP   DD      DD   SS           LL               TT
UU      UU   PP      PP   DD      DD   SS           LL               TT
UU      UU   PP      PP   DD      DD   SS           LL               TT
UU      UU   PP      PP   DD      DD   SS           LL               TT
UU      UU   PPPPPPPP     DD      DD     SSSSSS     LL               TT
UU      UU   PPPPPPPP     DD      DD     SSSSSS     LL               TT
UU      UU   PP           DD      DD           SS   LL               TT
UU      UU   PP           DD      DD           SS   LL               TT
UU      UU   PP           DD      DD           SS   LL               TT            ....
UU      UU   PP           DD      DD           SS   LL               TT            ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS     LLLLLLLLLL       TT            ....
UUUUUUUUUU   PP           DDDDDDDD     SSSSSSSS     LLLLLLLLLL       TT            ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
    428 000702  001412                          BEQ     10$
    429 000704                                  $PTNUM  #S$LRPR,,#SL$RPR
    430 000730  000207                  10$:    RETURN
    431
    432 000732  016700  000000G         SETBRA: MOV     .TEMP,R0
    433 000736  032760  000000G 000000G         BIT     #LF.BRO,S$LFLG(R0)
    434 000744  001412                          BEQ     10$
    435 000746                                  $PTNUM  #S$LBRA,,#SL$BRA
    436 000772  000207                  10$:    RETURN
    437
```

```
                                       .SBTTL  SETCST - SET THE STATION COST
                               ;+
                               ; *** - SETCST - SET THE STATION COST
                               ;
                               ; INPUTS:
                               ;       TEMP = TEMPLATE ADDRESS
                               ;       $OBUF = POINTER INTO OUTPUT BUFFER
                               ;
                               ; OUTPUTS:
                               ;       THE COST IS WRITTEN TO THE OUTPUT BUFFER
                               ;
                               ;-

000200                         SETCST:
000200  016702  177574                 MOV     TEMP,R2          ; GET THE TEMPLATE ADDRESS
000204  116201  000007                 MOVB    C.CST(R2),R1     ; GET THE STATION COST
000210  016700  000000G                MOV     $OBUF,R0         ; GET THE CURRENT POSITION IN THE OUTPUT BUFFER
000214  004767  000000G                CALL    $CNVD            ; CONVERT THE NUMBER TO DECIMAL ASCII
000220  010067  000000G                MOV     R0,$OBUF         ; UPDATE POINTER INTO OUTPUT BUFFER
000224  000207                         RETURN
```

```
    131                                                     .SBTTL  LOCAL DATA
    132
    133                                                     .NLIST  BEX
    134 000000       011       123       126    SVCDF:  .ASCII  <11>/SVC$DF/<11>/DLM-/
    135           000014                                SVCLN=.-SVCDF
    136 000014       117       116              ONSTA:  .ASCII  /ON/
    137           000002                                ONLEN=.-ONSTA
    138 000016       117       106       106    OFFSTA: .ASCII  /OFF/
    139           000003                                OFFLEN=.-OFFSTA
    140                                                 .LIST   BEX
    141                                                 .EVEN
    142
```

```
                                          .SBTTL  SETCWO - SETUP THE CHARACTERISTICS WORD 0
                                  ;+
                                  ; *** - SETCWO - SETUP THE CHARACTERISTICS WORD 0
                                  ;
                                  ; INPUT:
                                  ;       .PNUMB = THE CHARACTERISTICS WORD 0
                                  ;       .PNUMH = (HIGH ORDER)
                                  ;       .TEMP = START ADDRESS OF THE TEMPLATE
                                  ;
                                  ; OUTPUT:
                                  ;
                                  ;-

000136                            SETCWO:
000136  016702  000000G                   MOV     .TEMP,R2                ; GET START OF TEMPLATE
000142  016201  000006                    MOV     C.CWO(R2),R1    ; GET CHARACTERISTICS WORD 0
000146  016700  000000G                   MOV     $OBUF,R0        ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
000152  004767  000000G                   CALL    $CNVO           ; CONVERT NUMBER TO OCTAL
000156  010067  000000G                   MOV     R0,$OBUF        ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
000162  000207                            RETURN
```

```
                        .SBTTL  MACRO CALLS AND LOCAL DATA
                ;
                ; LOCAL MACROS
                ;

                        .MACRO  PRINT   TEXT
                        MOV     TEXT,R0         ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
                        CALL    $PRINT          ; PRINT MESSAGE
                        .ENDM   PRINT

                        .MACRO  ERROR$  TEXT
                ;
                ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
                ; MESSAGE STRING.
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                         ; GET ADDRESS OF ERROR MESSAGE
                        .ENDC
                        CALL    $CFERR                          ; PRINT ERROR MESSAGE
                        .ENDM   ERROR$

                        .MACRO  EPRINT  TEXT
                ;
                ; PRINT TEXT ON ERROR LUN
                ;
                        .IF DIF <TEXT><R0>
                        MOV     TEXT,R0                         ; GET ADDRESS OF ASCIZ STRING TO PRINT
                        .ENDC
                        CALL    $CFER1                          ; PRINT MESSAGE ON ERROR LUN
                        .ENDM   EPRINT
                ;
                ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
                ;
                        .MACRO  $$$PTN  ROU
                        .MACRO  'ROU    OFS,OCTL,OPT,?L
                        .IF NB <OPT>
                        MOV     OPT,R2
                        CALL    $PTOPT
                        BCS     L
                        .IFTF
                        MOV     OFS,R2
                        .IF NB <OCTL>
                        MOV     SP,R1
                        .IFF
                        CLR     R1
                        .ENDC ; IF NB <OCTL>
                        CALL    'ROU
                        .IFT  ; IF NB <OPT>
                L:
                        .ENDC ; IF NB <OPT>
                        .ENDM ; 'ROU
                        .ENDM ; $$$PTN

        000000          $$$PTN  $PTNUM
        000000          $$$PTN  $PTNMB
                ;
                ; PUT RAD50 TEMPLATE FIELD TO BUFFER
```

```
                                                 .SBTTL  $UPX29 - UPDATE X.29 CHARACTERISTICS
                                        ;+
                                        ; *** - $UPX29 - UPDATE X.29 CHARACTERISTICS
                                        ;
                                        ; INPUT:
                                        ;        R5 - ADDRESS OF TEMPLATE
                                        ;
                                        ; OUTPUT:
                                        ;        C-BIT = SUCCESS/FAILURE
                                        ;-

000010                                  $UPX29::
000010  012767  000000G 000000C                  MOV     #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
000016  010567  000000G                          MOV     R5,.TEMP              ; SAVE UPDATED TEMPLATE POINTER
000022  005725                                   TST     (R5)+                 ; POINT TO C.STS
000024  010567  000000G                          MOV     R5,.FLAGS             ; SAVE ADDRESS OF FLAGS WORD
000030  012767  177777  000000G                  MOV     #-1,.COMMA            ; SUPPRESS INITIAL COMMA
000036  012700  000000G                          MOV     #.BUFF,R0             ; POINT AT START OF OUTPUT BUFFER
000042  012701  000000'                          MOV     #X29DF,R1             ; POINT AT START OF X29 DEFINITION
000046  012702  000010                           MOV     #X29LN,R2             ; GET LENGTH OF X29 DEFINITION
000052  004767  000000G                          CALL    $MVASC                ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
000056  010067  000000G                          MOV     R0,$OBUF              ; SAVE CURRENT POSITION IN OUTPUT BUFFER
000062                                           $PTNUM  #X$9MXC               ; MAXIMUM CIRCUITS
000074                                           $PTNUM  #X$9CT                ; COUNTER TIMER
000106  016701  000000G                          MOV     $OBUF,R1              ; CALCULATE LENGTH OF BUFFER
000112  162701  000000G                          SUB     #.BUFF,R1             ;       ...
000116  010167  000000C                          MOV     R1,$OFDB+F.NRBD       ; SET LENGTH OF BUFFER
000122  004767  000000G                          CALL    $WDATA                ; WRITE THE RECORD TO THE OUTPUT FILE
000126  012767  000000G 000000C                  MOV     #$BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
000134  000207                                   RETURN


        000001                                   .END
```

N 12

```
MMM         MMM     LLL                 DDDDDDDDDDDD
MMM         MMM     LLL                 DDDDDDDDDDDD
MMM         MMM     LLL                 DDDDDDDDDDDD
MMMMMM   MMMMMM     LLL                 DDD         DDD
MMMMMM   MMMMMM     LLL                 DDD         DDD
MMMMMM   MMMMMM     LLL                 DDD         DDD
MMM   MMM   MMM     LLL                 DDD         DDD
MMM   MMM   MMM     LLL                 DDD         DDD
MMM   MMM   MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLL                 DDD         DDD
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD
MMM         MMM     LLLLLLLLLLLLLLL     DDDDDDDDDDDD
```

```
                                                  .SBTTL  DSPDRV  - Dispatch to Device Driver
                                          ;+
                                          ;                           *** DSPDRV ***
                                          ;
                                          ; Function:     To dispatch to a device driver's microcode load entry
                                          ;               point with a CCB initialized to point to one (or more)
                                          ;               buffer(s) containing microcode.
                                          ;
                                          ; Call Format:  CALL     DSPDRV
                                          ;
                                          ; Inputs:       R4       = address of CCB
                                          ;               PDVIDX   = device driver process index * 2
                                          ;               DDMTBL   = address of driver's line table
                                          ;
                                          ; Outputs:      Dispatch to driver is executed and buffer(s) loaded.
                                          ;               Device driver can expect R4 to point to the CCB but
                                          ;               must use KISAR6 to map to the buffer(s).
                                          ;
                                          ;               C-bit clear = load occurred without error
                                          ;               C-bit set   = load error(s) occurred
                                          ;               R3          = dispatch return status from driver
                                          ;
                                          ; Registers:    R3 modified for output
                                          ;
                                          ; Note:         $PDDSP saves KISAR5 and KISAR6 so that the driver does
                                          ;               not have to save/restore any APR5/APR6 mappings.
                                          ;-

 000604                                   DSPDRV: CALL    $SWSTK,20$              ;; Save registers, enter system state


                                          ;       If this is a write request,  then we must make sure that the load
                                          ;       CCB contains the correct mapping to the FCS buffer.  This mapping
                                          ;       can only be guaranteed to not change as long as MLD is in system
                                          ;       state.


 000610  126427  000011  000002                   CMPB    C.MOD(R4),#FS.WLB/400   ;; Is dispatch for a load?
 000616  001036                                   BNE     10$                     ;; Branch if not

                                                  .IFT                            ;; If system is RSX-11M ...

 000620  017764  177156  000014                   MOV     @ISAR5,C.BUF1(R4)       ;; Put file buffer mapping in CCB

                                                  .IFTF                           ;; For any operating system ...
                                                  .IF     DF      N$$UMR          ;; If system has UMRs ...

 000626  016401  000014                           MOV     C.BUF1(R4),R1           ;; Copy buffer's mapping bias
 000632  016402  000016                           MOV     C.BUF1+2(R4),R2         ;; Copy APR6-biased virtual address
 000636  062702  160000                           ADD     #160000,R2              ;; Force it to be APR5-biased
 000642  016400  000020                           MOV     C.CNT1(R4),R0           ;; Get buffer byte count
 000646  020077  177156                           CMP     R0,@RDBSZ               ;; Does microcode fit in an LDB?
 000652  003402                                   BLE     7$                      ;; Branch if so
 000654  017700  177150                           MOV     @RDBSZ,R0               ;; Else, copy one LDB's worth of code

 000660  010005                           7$:     MOV     R0,R5                   ;; Save the buffer length
 000662  016703  001366'                          MOV     LDBIAS,R3               ;; Get bias of UMR-mapped LDB
```

```
ABOAST  001152R  004 CL$XL2= 013700       C.FLG1  000022       D.MSK   000014       EV$SNA= 035000
AC$DNT= 000002       CL$XL3= 013600       C.FLG2  000032       D.NAM   000004       EV$SNF= 000516
AC$X25= 000001       CL$X2S= 013500       C.FNC   000010       D.PCB   000034       EV$SPE= 035001
AE$CIR= 000003       CM.CIR= 000002       C.LIN   000006       D.UCB   000002       EV$XCE= 034110
AE$LIN= 000001       CM.FMT= 100000       C.LNK   000000       D.UCBL  000010       EV$XDI= 013600
AE$MOD= 000004       CM.HRD= 000002       C.MOD   000011       D.UNIT  000006       EV$XGW= 034111
A$$CHK= 000000       CM.LIN= 000000       C.NSP   000004       D.VCAN= 000002       EV$XMX= 000514
A$$CPS= 000000       CM.LOO= 000001       C.PRO   000042       D.VDEB= 177776       EV$XRS= 000512
A$$PRI= 000000       CM.XLO= 000004       C.RSV   000002       D.VINI= 000000       EV$XSC= 000513
A$$TRP= 000000       CP.DCF= 000040       C.STA   000007       D.VOUT= 000004       EV$X2S= 013500
BLOCK   001372R  002 CP.HDL= 000007       C.STS   000012       D.VPWF= 000006       EV.CCB= 000001
BLXIO   000004RG 004 CP.PS = 177400       C.URM   177776       EF$ACT= 000001       EV.CIR= 000020
CB.CCB= 000002       CP.PSI= 000200       C.XACP  000004       ETYPCL= 010500       EV.LCB= 000100
CB.DDM= 000040       CP.XCF= 000100       C.XID   000035       EVTBLK  001526R  002 EV.LIN= 000004
CB.DLC= 000020       CP.2FR= 000030       C.XLEN  000044       EVTEFL= 000002       EV.MAP= 000002
CB.RDB= 000004       CS.ABO= 000100       C.XPLI  000040       EVTLUN= 000002       EV.MOD= 000040
CB.SDB= 000010       CS.BRO= 000002       C.XPT   000034       EV$ACF= 000201       EV.NOD= 000010
CB.SLI= 000100       CS.BUF= 000200       C.XSVC  000042       EV$ADR= 000420       EV.PRT= 000200
CB.XLB= 000001       CS.CES= 000002       C.XTC   000037       EV$ADU= 000417       EXEVCL= 000017
CCBRT   000016RG 004 CS.CHN= 000010       C.X25   000036       EV$APL= 000400       EXEVEC  000000R      004
CC.LLC= 000200       CS.CMP= 000200       DDMTBL  001404R  002 EV$ARC= 000421       E$DATA  000020
CE.ABO= 100362       CS.DCR= 000400       DEVERR  000051R  002 EV$AUC= 000010       E$EVTS  000000
CE.DAO= 100346       CS.DEF= 000004       DL$AST= 000002       EV$AUS= 000003       E$LCN   000016
CE.DIS= 100366       CS.DEV= 000002       DL$HLT= 000000       EV$CDF= 000520       E$LIN   000000
CE.ERR= 100370       CS.DIS= 000040       DL$IST= 000001       EV$COZ= 000011       E$MOD   000012
CE.ILN= 100350       CS.ENA= 000001       DL$MAI= 000004       EV$DBR= 000302       E$NOD   000010
CE.LTO= 100356       CS.ENB= 000020       DL$OFF= 000001       EV$GAS= 035101       E$PORT  000014
CE.MOP= 100372       CS.ERR= 100000       DL$ON = 000000       EV$HCE= 035114       E$PRM   000002
CE.NTE= 100361       CS.FTL= 001000       DL$RUN= 000003       EV$HCI= 035113       E$STAT  000006
CE.RTE= 100376       CS.HCR= 000001       DL$SHU= 000002       EV$HFE= 000506       E$TCB   000004
CE.SRC= 100364       CS.HFE= 002000       DL$SYN= 000005       EV$IFL= 000413       E$$XPR= 000000
CE.STP= 100352       CS.LST= 040000       DSDS    001312R  002 EV$IFO= 000415       E.CTL   000020
CE.TME= 100354       CS.MTL= 004000       DSPDRV  000604R  004 EV$IFS= 000414       E.DATA  000046
CE.TMO= 100374       CS.RNG= 000010       DV.CCL= 000002       EV$INF= 000515       E.EVT   000002
CE.UNS= 100344       CS.ROV= 000004       DV.COM= 020000       EV$LDL= 000407       E.LCN   000042
CF.CHN= 000001       CS.RSN= 010000       DV.DIR= 000010       EV$LDN= 010416       E.LEN   000216
CF.EOM= 000004       CS.SHU= 000001       DV.EXT= 000400       EV$LDO= 000411       E.LIN   000024
CF.HDR= 000020       CS.SID= 000002       DV.F11= 040000       EV$LDS= 000410       E.LNK   000000
CF.LB = 100000       CS.STR= 000004       DV.ISP= 002000       EV$LSC= 000500       E.MOD   000036
CF.LIN= 000002       CS.SUC= 000001       DV.MBC= 000400       EV$LUP= 000412       E.NOD   000034
CF.SOM= 000010       CS.TMO= 020000       DV.MNT= 100000       EV$NOL= 000402       E.PDV   000021
CF.SYN= 000040       CS.XUR= 000004       DV.MSD= 000100       EV$NRC= 000416       E.PORT  000040
CF.TRN= 000100       CTLNUM  001402R  002 DV.OSP= 004000       EV$NSC= 000200       E.PRM   000026
CL$ASZ= 010500       CURQST  001376R  002 DV.PSE= 010000       EV$NUL= 000401       E.PVC   000044
CL$DLL= 000500       CVTPRM= 014010       DV.REC= 000001       EV$NVR= 000406       E.SIZ   000022
CL$ECL= 000300       C$$CKP= 000000       DV.SDJ= 000020       EV$OPL= 000403       E.TIME  000004
CL$LDN= 010400       C$$ORE= 000400       DV.SQD= 000040       EV$PCC= 034000       FCSBSZ= 001000
CL$MAN= 000000       C$$RSH= 177564       DV.SWL= 001000       EV$PCI= 034001       FCSBUF  000304R      002
CL$PAZ= 034100       C.ADD   000034       DV.TTY= 000004       EV$PCM= 034002       FCSEFL= 000001
CL$PLH= 034000       C.BID   000003       DV.UMD= 000200       EV$PFE= 000404       FCSERR  001302R      004
CL$PLL= 000600       C.BUF   000014       D$$BUG= 177514       EV$PPC= 034003       FCSIOS  001304R      002
CL$PRT= 034200       C.BUF1  000014       D$$ISK= 000000       EV$RCF= 000517       FCSLUN= 000001
CL$ROU= 010000       C.BUF2  000024       D$$L11= 000001       EV$RDC= 010001       FC.CCP= 000020
CL$SES= 000200       C.CNT   000020       D$$YNC= 000000       EV$RDR= 010002       FC.CTL= 000006
CL$SGE= 035000       C.CNT1  000020       D$$YNM= 000000       EV$RJE= 035106       FC.KCP= 000016
CL$SSE= 035100       C.CNT2  000030       D.DSP   000012       EV$RSC= 000501       FC.KIL= 000004
CL$TRN= 000400       C.FLG   000022       D.LNK   000000       EV$RUL= 000405       FC.MAN= 000024
```

```
MLD        CREATED BY  MACRO  ON 28-AUG-85 AT 11:27      PAGE 10
MACRO CROSS REFERENCE                                    CREF   04.00
MACRO NAME        REFERENCES

STSE$             #6-89       8-203
TCBDF$            #6-79       6-81
UCBDF$            #6-65       6-67
WAIT$             #6-64       15-796
```

B	1	
C	1	
D	1	
E	1	
F	1	UPDPVC
G	1	
H	1	
I	1	
J	1	
K	1	
L	1	
M	1	
N	1	
B	2	
C	2	
D	2	UPDRDT
E	2	
F	2	
G	2	
H	2	
I	2	
J	2	
K	2	
L	2	
M	2	
N	2	UPDREM
B	3	
C	3	
D	3	
E	3	
F	3	
G	3	
H	3	
I	3	
J	3	
K	3	
L	3	
M	3	
N	3	
B	4	
C	4	UPDROU
D	4	
E	4	
F	4	
G	4	
H	4	
I	4	
J	4	
K	4	
L	4	
M	4	
N	4	UPDSER
B	5	
C	5	
D	5	
E	5	
F	5	
G	5	
H	5	
I	5	
J	5	
K	5	
L	5	
M	5	
N	5	UPDSLT
B	6	
C	6	
D	6	
E	6	
F	6	
G	6	
H	6	
I	6	
J	6	
K	6	
L	6	
M	6	
N	6	
B	7	
C	7	
D	7	
E	7	
F	7	
G	7	UPDSTA
H	7	
I	7	
J	7	
K	7	
L	7	
M	7	
N	7	
B	8	
C	8	
D	8	
E	8	
F	8	
G	8	
H	8	
I	8	UPDSVC
J	8	
K	8	
L	8	
M	8	
N	8	
B	9	
C	9	
D	9	
E	9	
F	9	UPDUNT
G	9	
H	9	
I	9	
J	9	
K	9	
L	9	
M	9	
N	9	
B	10	
C	10	
D	10	
E	10	
F	10	
G	10	
H	10	
I	10	
J	10	
K	10	UPDX2P
L	10	
M	10	
N	10	
B	11	
C	11	
D	11	
E	11	
F	11	
G	11	
H	11	UPDX29
I	11	
J	11	
K	11	
L	11	
M	11	
N	11	
B	12	
C	12	
D	12	
E	12	UPDX3P
F	12	
G	12	
H	12	
I	12	
J	12	
K	12	
L	12	
M	12	
N	12	
B	13	MLD
C	13	
D	13	
E	13	
F	13	
G	13	
H	13	
I	13	
J	13	
K	13	
L	13	
M	13	
N	13	
B	14	
C	14	
D	14	
E	14	
F	14	
G	14	
H	14	
I	14	
J	14	
K	14	
L	14	
M	14	
N	14	
B	15	
C	15	
D	15	
E	15	
F	15	
G	15	
H	15	
I	15	
J	15	
K	15	
L	15	
M	15	
N	15	
B	16	
C	16	MLD11S
D	16	
E	16	
F	16	
G	16	
H	16	
I	16	
J	16	
K	16	
L	16	
M	16	

```
                                              STAF$S  K.URM(R2)               ; Set new CPU/UNIBUS affinity

                                              .ENDC                           ; End RSX-11M+ conditional



                                      ;       Now locate the microcode in the system.  If the microcode has been
                                      ;       successfully located,  perform the load cycle.

     000312  062704  000014           60$:    ADD     #C.BUF1,R4              ; Point at microcode ASCII name length
     000316  012405                           MOV     (R4)+,R5                ; Copy it and point at start of name

                                              .IFT                            ; If system is RSX-11M ...

                                              CALL    LOCFIL                  ; Locate microcode file

                                              .IFF                            ; If system is RSX-11S ...

     000320                                   CALL    LOCPAR                  ; Locate microcode partition

                                              .IFTF                           ; For any operating system ...

     000324  016704  000156'                  MOV     CURQST,R4               ; Recover CCB address
     000330  103003                           BCC     70$                     ; Branch if microcode was found
     000332  012703  001000                   MOV     #CS.FTL,R3              ; Else, get "fatal error" return
     000336  000407                           BR      80$                     ; Branch to common code

     000340  012764  001026  000010   70$:    MOV     #FC.MLD+FS.WLB,C.FNC(R4); Change CCB (sub)function

                                              .IFT                            ; If system is RSX-11M ...

                                              CALL    MLOAD                   ; Load microcode from disk

                                              .IFF                            ; If system is RSX-11S ...

     000346                                   CALL    SLOAD                   ; Load microcode from partition

                                              .IFTF                           ; For any operating system ...

     000352  005703                           TST     R3                      ; Did the driver stop the load?
     000354  100672                           BMI     10$                     ; Branch if so, save a dispatch

     000356  016704  000156'  000010  80$:    MOV     CURQST,R4               ; Get address of current CCB
     000362  012764  000026   000010          MOV     #FC.MLD+FS.HLT,C.FNC(R4); Set CCB function to "load halted"
     000370  052703  100000                   BIS     #CS.ERR,R3              ; "OR" general error flag into status
     000374  010364  000012                   MOV     R3,C.STS(R4)            ; Pass load status
     000400                                   CALL    DSPDRV                  ; Dispatch to device driver
     000404  000656                           BR      10$                     ; Look for another load request

     000406                           90$:    EXIT$S                          ; Program exit
```

```
    988                                       .ENDC                           ; End RSX-11S conditional
    989          000000'                      .END    MLD                     ; End of source file
```

```
MMM        MMM        AAAAAAAAA       NNN        NNN       AAAAAAAAA          CCCCCCCCCCCC    PPPPPPPPPPPP
MMM        MMM        AAAAAAAAA       NNN        NNN       AAAAAAAAA          CCCCCCCCCCCC    PPPPPPPPPPPP
MMM        MMM        AAAAAAAAA       NNN        NNN       AAAAAAAAA          CCCCCCCCCCCC    PPPPPPPPPPPP
MMMMMM  MMMMMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP        PPP
MMMMMM  MMMMMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP        PPP
MMMMMM  MMMMMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP        PPP
MMM   MMM  MMM     AAA        AAA     NNNNNN     NNN    AAA        AAA     CCC                PPP        PPP
MMM   MMM  MMM     AAA        AAA     NNNNNN     NNN    AAA        AAA     CCC                PPP        PPP
MMM   MMM  MMM     AAA        AAA     NNNNNN     NNN    AAA        AAA     CCC                PPP        PPP
MMM        MMM     AAA        AAA     NNN   NNN  NNN    AAA        AAA     CCC                PPPPPPPPPPPP
MMM        MMM     AAA        AAA     NNN   NNN  NNN    AAA        AAA     CCC                PPPPPPPPPPPP
MMM        MMM     AAA        AAA     NNN   NNN  NNN    AAA        AAA     CCC                PPPPPPPPPPPP
MMM        MMM     AAAAAAAAAAAAAA     NNN     NNNNNN    AAAAAAAAAAAAAA     CCC                PPP
MMM        MMM     AAAAAAAAAAAAAA     NNN     NNNNNN    AAAAAAAAAAAAAA     CCC                PPP
MMM        MMM     AAAAAAAAAAAAAA     NNN     NNNNNN    AAAAAAAAAAAAAA     CCC                PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA     CCC                PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA        CCCCCCCCCCCC    PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA        CCCCCCCCCCCC    PPP
MMM        MMM     AAA        AAA     NNN        NNN    AAA        AAA        CCCCCCCCCCCC    PPP
```

```
**FILE**ID**MANACP
```

```
    477                                                 .SBTTL  RETURN I/O ERRORS
    478                                         ;+
    479                                         ; THESE ROUTINES RETURN I/O ERROR CODES FOR $IOFIN COMPLETION.
    480                                         ;
    481                                         ; INPUTS:
    482                                         ;       NONE
    483                                         ;
    484                                         ; OUTPUTS:
    485                                         ;       R0 = I/O ERROR CODE
    486                                         ;       R1 = ZERO
    487                                         ;
    488                                         ; REGISTERS:
    489                                         ;       ALL OTHER REGISTERS ARE PRESERVED
    490                                         ;-
    491
    492                                                 .ENABL  LSB
    493
    494 002226  012700  000376                  IFCERR: MOV     #IE.IFC&377,R0          ; ILLEGAL FUNCTION
    495 002232  000416                                  BR      10$
    496
    497 002234  012700  000336                  ALNERR: MOV     #IE.ALN&377,R0          ; FILE ALREADY ACCESSED ON LUN
    498 002240  000413                                  BR      10$
    499
    500 002242  012700  000363                  DAOERR: MOV     #IE.DAO&377,R0          ; DATA OVERRUN
    501 002246  000410                                  BR      10$
    502
    503 002250  012700  000270                  NDRERR: MOV     #IE.NDR&377,R0          ; NO DYNAMIC SPACE AVAILABLE
    504 002254  000405                                  BR      10$
    505
    506 002256  012700  000333                  NLNERR: MOV     #IE.NLN&377,R0          ; NO FILE ON LUN
    507 002262  000402                                  BR      10$
    508
    509 002264  012700  000360                  PRIERR: MOV     #IE.PRI&377,R0          ; PRIVILEGE VIOLATION
    510
    511 002270  005267  175722                  10$:    INC     DELFLG                  ; MARK CONTEXT FOR DELETION
    512 002274  005001                                  CLR     R1                      ; NO BYTES TO RETURN
    513 002276                                          RETURN
    514
    515                                                 .DSABL  LSB
```

```
                                                                        B  5
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

DAOERR     002242 R         6-402       6-432       #7-500
DDEACB     000016 RG       #5-91
DDMAN      000066 RG       #5-111
DEACTX  =  ******   GX      6-437
DECPT      000070 RG       #5-112
DELFLG     000216 R        #5-263      *6-315        6-435      *7-511      *8-594      *9-626      *10-651
DREXT      000020 RG       #5-92        6-366
EXRQN      000022 RG       #5-93
EXRQP      000024 RG       #5-94
FDNCNT     000224 RG       #5-266      *6-337        6-358
FE.CEX  =  ******   GX      6-324
FILHD      000072 RG       #5-113
FINREQ     002736 R         8-557       8-595        9-627      #11-673
FMASK      000026 RG       #5-95        6-324
FRKDON  =  ******   GX      6-412
FTABLE  =  ******   GX      5-272
GETPKT     001502 RG       #6-312       6-413
HEADR      000030 RG       #5-96
IDBUFF     001410 RG       #5-277       8-582
IE.ALN  =  177736           7-497
IE.DAO  =  177763           7-500
IE.IFC  =  177776           7-494
IE.NDR  =  177670           7-503
IE.NLN  =  177733           7-506
IE.PRI  =  177760           7-509
IFCERR     002226 R         6-475      #7-494
INIACP  =  ******   GX      6-307
INIREQ  =  ******   GX      8-589
IOFIN      000032 RG       #5-97        6-467
ISAR5      000002 RG       #5-85
ISAR6      000004 RG       #5-86
ISTACK     000226 RG       #5-267      *6-306
IS.SUC  =  000001           10-652      11-674
I.FCN      000012  G        6-395       6-417        8-573       8-576
I.LN2      000006  G        6-387       6-463
I.PRM      000024  G        6-397       6-400        6-430      *6-439       8-541       8-554      *8-554       8-555      *8-555
                            8-556      *8-556        8-580       8-583       8-584      *8-584       8-585      *8-585       8-586
                           *8-586       9-620
I.TCB      000004  G        6-351       6-371       *6-372      *6-443       8-561
KISAR5  =  ******   GX      5-85
KISAR6  =  ******   GX      5-86        5-87
KSAR6      000006 RG       #5-87
LDBAF      000074 RG       #5-114
LDBGT      000076 RG       #5-115
LDBRT      000100 RG       #5-116
LGCON      000102 RG       #5-117
LGFNB      000104 RG       #5-118
LGMON      000106 RG       #5-119
LGSTT      000110 RG       #5-120
LGUIC      000112 RG       #5-121
LLCRS      000114 RG       #5-122
LLCTA      000116 RG       #5-123
```

```
                                                                        C  5
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 3
```

```
                                                                   B  6
MANAST     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

ASTX        000022 R        #6-86       8-143
ASTX1       000024 R        #6-87       7-118
CF$FDN   = ******   GX       6-84       7-116        8-142
C$FLAG   = ******   GX      *6-84      *7-116       *8-142
C$LOC    = ******   GX      *8-142
C$STAT   = ******   GX      *6-85      *7-116
DLXAST      000054 RG       #7-113
EXRQN    = ******   GX       6-90
FDNCNT   = ******   GX      *6-91
MRKAST      000074 RG       #8-139
NTLAST      000000 RG       #6-81
REXAST      000046 RG       #6-92
TKTCB    = ******   GX       6-89
```

```
                                                                   C  6
MANAST     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 2
```

```
                                  .SBTTL  WBUFF - WRITE "BUFFER" INTO USER'S BUFFER
                          ;+
                          ; **-WBUFF-WRITE "BUFFER" INTO USER'S BUFFER
                          ;
                          ; THIS ROUTINE COPIES THE CONTENTS OF "BUFFER" INTO THE USER'S BUFFER.
                          ;
                          ; INPUTS:
                          ;       R1 = LENGTH OF DATA TO COPY
                          ;       R3 = ADDRESS OF I/O PACKET WITH:
                          ;               I.PRM   = APR BIAS OF BUFFER
                          ;               I.PRM+2 = VIRTUAL ADDRESS OF BUFFER
                          ;
                          ; OUTPUTS:
                          ;       DATA COPIED TO USER'S BUFFER
                          ;
                          ; REGISTERS:
                          ;       NO REGISTERS MODIFIED
                          ;-

000526                    WBUFF::
000526                            CALL    $SAVAL                  ;; SAVE ALL REGISTERS
000532  010146                    MOV     R1,-(SP)                ;; SAVE NUMBER OF BYTES TO MOVE
000534  012700  000000G           MOV     #BUFFER,R0              ;; GET DESTINATION BUFFER ADDRESS
000540                            CALL    @RELOC                  ;; RELOCATE THE BUFFER          ; RJK02
000544  042702  160000            BIC     #160000,R2              ;; STRIP OFF APR OFFSET
000550  052702  120000            BIS     #120000,R2              ;; MAKE SOURCE DISPLACEMENT VIA APR 5
000554  012600                    MOV     (SP)+,R0                ;; RETRIEVE NUMBER OF BYTES TO MOVE
000556  016304  000026            MOV     I.PRM+2(R3),R4          ;; GET DESTINATION APR 6 DISPLACEMENT
000562  016303  000024            MOV     I.PRM(R3),R3            ;; GET DESTINATION APR 6 BIAS
000566                            CALLR   @BLXIO                  ;; COPY DATA AND RETURN         ; RJK02

        000001                    .END
```

```
                                        ;       NEW ENTRIES CAN BE ADDED BY DEFINING THEIR ADDRESS IN THE PSECT "$$$BYE",
                                        ;       AND LINKING THE MODULES IN THE SAME OVERLAY SEGMENT (BUT AFTER!) MANBYE.
                                        ;
                                        ;       EACH LOGOUT ROUTINE IS CALLED WITH:
                                        ;               R3 = ADDRESS OF BUFFER
                                        ;               R4 = ADDRESS OF CONTEXT AREA
                                        ;       ROUTINES SHOULD PRESERVE R4 AND R5, BUT CAN MODIFY R0 - R3.
                                        ;
000000                                          .PSECT  $$$BYE
000000  000040'                         BYETBL: .WORD   DECBYE                  ; CLEANUP FOR DECNET
000000                                          .PSECT  $$$BYF                  ; FORCE NEXT PSECT TO BE CONTIGUOUS
000000  000000                                  .WORD   0                       ; END-OF-TABLE MARKER
000000                                          .PSECT                          ; RETURN TO CODE PSECT
```

```
                                        .SBTTL  FRKDON - CHECK FOR COMPLETED FORKS AND RE-LOAD CONTEXT
                                ;+
                                ; **-FRKDON-CHECK FOR COMPLETED FORKS AND RE-LOAD CONTEXT
                                ;
                                ; THIS ROUTINE IS CALLED TO CHECK IF ANY CONTEXT FORKS ARE COMPLETED.
                                ; IF NONE ARE, THE ROUTINE RETURNS, ELSE, THE CONTEXT IS LOADED, AND
                                ; THE THREAD IS RESUMED.
                                ;
                                ; INPUTS:
                                ;       CTXHD =  CONTEXT LISTHEAD
                                ;
                                ; OUTPUTS:
                                ;       IF NO FORKS ARE COMPLETE, NONE.
                                ;       ELSE:
                                ;         CURCTX = ADDRESS OF SAVED CONTEXT AREA WITH
                                ;         CURIOP = ADDRESS OF CURRENT I/O PACKET
                                ;         STATUS = C$STAT (RETURN STATUS) FROM SAVED CONTEXT
                                ;         BUFFER = BUFFER COPIED FROM USER'S BUFFER
                                ;         REGISTERS RESTORED.
                                ;
                                ;-

000202                          FRKDON::
                                ;
                                ; LOOK FOR FIRST COMPLETED FORK
                                ;
000202  012700  000000G                 MOV     #CTXHD,R0               ; POINT TO CONTEXT AREA LISTHEAD
000206  011000                  10$:    MOV     (R0),R0                 ; GET ADDRESS OF NEXT CONTEXT AREA
000210  001001                          BNE     20$                     ; IF NE, GOT ONE
000212                                  RETURN                          ; ELSE, NO FORK TO PROCESS - RETURN

000214  032760  000000G 000000G 20$:    BIT     #CF$FDN,C$FLAG(R0)      ; IS THERE A COMPLETED FORK HERE ?
000222  001771                          BEQ     10$                     ; IF EQ, NO - KEEP LOOKING
                                ;
                                ; RELOAD TASK STACK, RESTORE CURRENT I/O PACKET AND "TI:" ADDRESS
                                ;
000224  042760  000000G 000000G         BIC     #CF$FDN,C$FLAG(R0)      ; CLEAR FORK DONE FLAG
000232  005367  000000G                 DEC     FDNCNT                  ; THIS FORK NO LONGER NEEDS PROCESSING
000236  010067  000000G                 MOV     R0,CURCTX               ; SET THIS AS CURRENT CONTEXT
000242                                  CALL    UPDCTX                  ; RESTORE THE CONTEXT
000246  016706  000000G                 MOV     ISTACK,SP               ; RE-INITIALIZE TASK STACK
000252  016001  000000G                 MOV     C$SP(R0),R1             ; AND GET SAVED THREAD SP
000256  012146                  30$:    MOV     (R1)+,-(SP)             ; RELOAD TASK STACK
000260  020127  000000C                 CMP     R1,#CONTEX+CTXSIZ       ; AT END OF CONTEXT STACK ?
000264  001374                          BNE     30$                     ; IF NE, NO
000266  017701  000000G                 MOV     @TKTCB,R1               ; GET OUR TCB ADDRESS          ; RJK01
000272  012661  000026                  MOV     (SP)+,T.UCB(R1)         ; ... AND RESTORE OUR TI: UCB ADDRESS
000276  012667  000000G                 MOV     (SP)+,TSKLN2            ; RESTORE TASK'S 2ND LUN WORD
000302  012667  000000G                 MOV     (SP)+,CURIOP            ; RESTORE AND I/O PACKET ADDRESS
000306  016067  000000G 000000G         MOV     C$STAT(R0),STATUS       ; RETURN STATUS
                                ;
                                ; RELOAD THE OVERLAY SEGMENTS
                                ;
000314  012605                          MOV     (SP)+,R5                ; GET THE NUMBER OF SEGMENTS TO LOAD
000316  005305                  40$:    DEC     R5                      ; ANY MORE TO LOAD ?
000320  002406                          BLT     60$                     ; IF LT, NO - FINISHED
000322  012667  000004                  MOV     (SP)+,50$               ; ELSE, POINT TO SEGMENT DESCRIPTOR
000326                                  CALL    @.NAUTO                 ; CALL $AUTO ...
```

```
                                        SEC                             ; ELSE, SET CARRY

                                        .ENDC   ; DF R$$PRO

 000402  016702  177404         89$:    MOV     NMSTAT+2,R2             ; GET BUFFER LENGTH
 000406  001003                         BNE     90$                     ; IF NE, SOMETHING IN BUFFER
 000410  005202                         INC     R2                      ; ELSE, INDICATE ONE BYTE
 000412  100001                         BPL     90$                     ; IF PL, SUCCESS
 000414  000261                         SEC                             ; ELSE, ERROR
 000416                         90$:    RETURN

                                        .DSABL  LSB

```

```
**FILE**ID**MANTRA
```

```
MM      MM     AAAAAA     NN      NN   TTTTTTTTTT   RRRRRRRR       AAAAAA
MM      MM     AAAAAA     NN      NN   TTTTTTTTTT   RRRRRRRR       AAAAAA
MMMM  MMMM   AA      AA   NN      NN       TT       RR      RR   AA      AA
MMMM  MMMM   AA      AA   NN      NN       TT       RR      RR   AA      AA
MM  MM  MM   AA      AA   NNNN    NN       TT       RR      RR   AA      AA
MM  MM  MM   AA      AA   NNNN    NN       TT       RR      RR   AA      AA
MM      MM   AA      AA   NN  NN  NN       TT       RRRRRRRR     AA      AA
MM      MM   AA      AA   NN  NN  NN       TT       RRRRRRRR     AA      AA
MM      MM   AAAAAAAAAA   NN    NNNN       TT       RR  RR       AAAAAAAAAA
MM      MM   AAAAAAAAAA   NN    NNNN       TT       RR  RR       AAAAAAAAAA
MM      MM   AA      AA   NN      NN       TT       RR    RR     AA      AA      ....
MM      MM   AA      AA   NN      NN       TT       RR    RR     AA      AA      ....
MM      MM   AA      AA   NN      NN       TT       RR      RR   AA      AA      ....
MM      MM   AA      AA   NN      NN       TT       RR      RR   AA      AA      ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31
Table of contents
```

B 12

```
MANTRA    CREATED BY  MACRO  ON 29-JUN-85 AT 10:31     PAGE 2
MACRO CROSS REFERENCE                                  CREF   04.00

MACRO NAME       REFERENCES

ANBDF$           #5-54       5-57
CALL              6-90       7-122      8-155      9-191      9-193
MANDF$           #5-54       5-56
RESMAP           #5-54       9-213
RETC             #5-54       9-195
RETURN            6-96       7-129      8-164      9-214      9-230
SAVMAP           #5-54       9-192
SWSTK$            9-191
```

C 12

```
**FILE**ID**MANVCD
```

Table of contents

```
     5-   50    MACRO CALLS AND LOCAL DEFINITIONS
     6-   72    FTABLE - FUNCTION DISPATCH TABE
```

Table of contents

```
     5-   50     Macro calls and definitions
     6-   63     PSIBYE - LOGOUT cleanup for PSI user
```

```
                                  ; IF "CLEAR" OPERATION FOR SPECIFIED ALIAS, MAKE SURE THAT IT EXISTS
                                  ;
000102  032764  000100  000002            BIT     #MO$CLE,A$OPT(R4)       ; IS THIS A CLEAR OPERATION ?
000110  001432                            BEQ     10$                     ; IF EQ, NO - BYPASS CHECK FOR EXISTANCE
000112  105764  000011                    TSTB    A$TYP(R4)               ; WAS AN ALIAS NAME SPECIFIED ?
000116  003432                            BLE     20$                     ; IF LE, NO - BYPASS CHECK
000120  010400                            MOV     R4,R0                   ; ELSE, POINT TO CONTEXT BLOCK
000122  062700  000012                    ADD     #A$ALI,R0               ; POINT TO ALIAS NAME
000126  016401  000020                    MOV     A$SCO(R4),R1            ; GET THE SCOPE
000132                                    CALL    TSTALI                  ; TRY TO FIND THE ALIAS
000136  012700  177770                    MOV     #ME.UCO,R0              ; ASSUME IT DIDN'T EXIST
000142  012764  000010  000004            MOV     #ME$ALI,A$PAR(R4)       ; ASSUME VERSION 2.1 OR HIGHER
000150  016703  000000G                   MOV     CURCTX,R3               ; GET ADDRESS OF CONTEXT AREA
000154  032763  000000G 000000G           BIT     #CF$VR2,C$FLAG(R3)      ; CONNECTED TO VERSION 2 NCP?
000162  001403                            BEQ     5$                      ; BR IF NO
000164  012764  000003  000004            MOV     #ME$AL2,A$PAR(R4)       ; ELSE STORE VERSION 2 ENTITY CODE
000172  103412                    5$:     BCS     30$                     ; IF CS, INVALID ALIAS ID
000174  000403                            BR      20$                     ; ELSE, IT EXISTED
                                  ;
                                  ; IF "SET" OPERATION CHECK DESTINATION
                                  ;
000176                            10$:    CALL    CHKDES                  ; ELSE, CHECK DESTINATION
000202  103406                            BCS     30$                     ; IF CS, ILLEGAL DESTINATION
                                  ;
                                  ; CHECK FOR JUNK AT END OF MESSAGE
                                  ;
000204  012700  177776            20$:    MOV     #ME.FOR,R0              ; ASSUME ILLEGAL INPUT FORMAT
000210  005702                            TST     R2                      ; ANYTHING LEFT IN INPUT BUFFER ?
000212  001002                            BNE     30$                     ; IF NE, YES - ILLEGAL INPUT
000214  012700  000002                    MOV     #MS.MOR,R0              ; ELSE, SET SUCCESSFUL RETURN
                                  ;
                                  ; USE THE STATUS IN R0 TO BUILD THE RETURN MESSAGE AND C-BIT RETURN
                                  ;
000220  012702  000001            30$:    MOV     #1,R2                   ; ASSUME 1 BYTE RETURN MESSAGE
000224  016403  000006                    MOV     A$BUF(R4),R3            ; RESTORE THE BUFFER ADDRESS
000230  052764  000001  000000            BIS     #AF.INI,A$FLG(R4)       ; ASSUME INIT WAS SUCCESSFUL
000236  110013                            MOVB    R0,(R3)                 ; SET THE RETURN CODE
000240  100013                            BPL     50$                     ; IF PL, SUCCESS - FINISHED
000242  005064  000000                    CLR     A$FLG(R4)               ; ELSE, UNSUCCESSFUL INIT
000246  116463  000004  000001            MOVB    A$PAR(R4),1(R3)         ; SET THE PARAMETER TYPE IN ERROR
000254  001404                            BEQ     40$                     ; IF EQ, THERE WAS NONE
000256  105063  000002                    CLRB    2(R3)                   ; ELSE, MAKE PARAMETER TYPE 2 BYTES LONG
000262  012702  000003                    MOV     #3,R2                   ; SET LENGTH OF RETURN TO 3 BYTES
000266  000261                    40$:    SEC                             ; SET ERROR RETURN
000270                            50$:    RETURN
```

```
                                            .SBTTL  ALCHOP - ALIAS CHANGE PARAMETERS OPERATE
                                    ;+
                                    ;**-ALCH  -ALIAS CHANGE PARAMETERS OPERATE ROUTINE
                                    ;
                                    ; THIS ROUTINE IS CALLED TO PERFORM THE CHANGE ALIAS PARAMETERS
                                    ; OPERATIONS (SET AND CLEAR).
                                    ;
                                    ; INPUTS:
                                    ;         R3 = ADDRESS OF INPUT/OUTPUT BUFFER
                                    ;         R4 = ADDRESS OF CONTEXT AREA (WITH DATA SET UP FROM ALCHIN)
                                    ;
                                    ; OUTPUTS:
                                    ;         R0 = NICE RETURN CODE
                                    ;         R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                    ;         R3 = ADDRESS OF OUTPUT BUFFER
                                    ;         C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                    ;
                                    ; REGISTERS:
                                    ;         R3, R4, R5 ARE PRESERVED
                                    ;
                                    ;-
000000                              ALCHOP::
000000  004567  000000G                     JSR     R5,$SAVRG               ; SAVE R3-R5
000004  010364  000006                      MOV     R3,A$BUF(R4)            ; SAVE BUFFER POINTER
000010  012700  177600                      MOV     #ME.DON,R0              ; ASSUME WE'RE ALL DONE
000014  032764  000001  000000              BIT     #AF.INI,A$FLG(R4)       ; HAVE WE BEEN INITIALIZED ?
000022  001411                              BEQ     20$                     ; IF EQ, NO - ALL DONE !
                                    ;
                                    ; DISPATCH TO SET OR CLEAR ROUTINES
                                    ;
000024  032764  000100  000002              BIT     #MO$CLE,A$OPT(R4)       ; IS THIS A CLEAR OPERATION ?
000032  001403                              BEQ     10$                     ; IF EQ, NO - SET
000034                                      CALL    CLRALI                  ; ELSE, YES - CLEAR ALIAS(ES)
000040  000402                              BR      20$                     ; AND RETURN
000042                              10$:    CALL    SETALI                  ; SET ALIAS
                                    ;
                                    ; USE THE STATUS IN R0 TO BUILD RETURN MESSAGE AND C-BIT RETURN
                                    ;
000046  005064  000000              20$:    CLR     A$FLG(R4)               ; CLEAR INIT FLAG IN FLAGS WORD
000052  012702  000001                      MOV     #1,R2                   ; MESSAGE IS 1 BYTE LONG
000056  016403  000006                      MOV     A$BUF(R4),R3            ; RESTORE BUFFER POINTER
000062  110013                              MOVB    R0,(R3)                 ; BUILD THE RETURN MESSAGE
000064  100001                              BPL     30$                     ; IF PL, SUCCESS
000066  000261                              SEC                             ; ELSE, FAILURE
000070                              30$:    RETURN
```

```
                                        .SBTTL  DSPDRV  - Dispatch to Device Driver
                                ;+
                                                        *** DSPDRV ***
                                ;
                                ; Function:     To dispatch to a device driver's microcode load entry
                                ;               point with a CCB initialized to point to one (or more)
                                ;               buffer(s) containing microcode.
                                ;
                                ; Call Format:  CALL    DSPDRV
                                ;
                                ; Inputs:       R4      = address of CCB
                                ;               PDVIDX  = device driver process index * 2
                                ;               DDMTBL  = address of driver's line table
                                ;
                                ; Outputs:      Dispatch to driver is executed and buffer(s) loaded.
                                ;               Device driver can expect R4 to point to the CCB but
                                ;               must use KISAR6 to map to the buffer(s).
                                ;
                                ;               C-bit clear = load occurred without error
                                ;               C-bit set   = load error(s) occurred
                                ;               R3          = dispatch return status from driver
                                ;
                                ; Registers:    R3 modified for output
                                ;
                                ; Note:         $PDDSP saves KISAR5 and KISAR6 so that the driver does
                                ;               not have to save/restore any APR5/APR6 mappings.
                                ;-

000414                          DSPDRV: CALL    $SWSTK,20$              ;; Save registers, enter system state


                                ;       If this is a write request,  then we must make sure that the load
                                ;       CCB contains the correct mapping to the FCS buffer.  This mapping
                                ;       can only be guaranteed to not change as long as MLD is in system
                                ;       state.

000420  126427  000011  000002          CMPB    C.MOD(R4),#FS.WLB/400   ;; Is dispatch for a load?
000426  001033                          BNE     10$                     ;; Branch if not

                                        .IFT                            ;; If system is RSX-11M ...

                                        MOV     @ISAR5,C.BUF1(R4)       ;; Put file buffer mapping in CCB

                                        .IFTF                           ;; For any operating system ...
                                        .IF     DF      N$$UMR          ;; If system has UMRs ...

000430  016401  000014                  MOV     C.BUF1(R4),R1           ;; Copy buffer's mapping bias
000434  016402  000016                  MOV     C.BUF1+2(R4),R2         ;; Copy APR6-biased virtual address
000440  062702  160000                  ADD     #160000,R2              ;; Force it to be APR5-biased
000444  016400  000020                  MOV     C.CNT1(R4),R0           ;; Get buffer byte count
000450  020077  000616'                 CMP     R0,@RDBSZ               ;; Does microcode fit in an LDB?
000454  003402                          BLE     7$                      ;; Branch if so
000456  017700  000616'                 MOV     @RDBSZ,R0               ;; Else, copy one LDB's worth of code

000462  010005                  7$:     MOV     R0,R5                   ;; Save the buffer length
000464  016703  000146'                 MOV     LDBIAS,R3               ;; Get bias of UMR-mapped LDB
```

```
Symbol table

ABOAST  000754R  003 CL$DLL= 000500       C$$CKP= 000000       EV$APL= 000400       EXEVEC  000566R  002
AC$DNT= 000002       CL$ECL= 000300       C$$ORE= 000400       EV$ARC= 000421       E$DATA  000020
AC$X25= 000001       CL$LDN= 010400       C$$RSH= 177564       EV$AUC= 000010       E$EVTS  000000
AE$CIR= 000003       CL$MAN= 000000       C.ADD   000034       EV$AUS= 000003       E$LCN   000016
AE$LIN= 000001       CL$PAZ= 034100       C.BID   000003       EV$CDF= 000520       E$LIN   000000
AE$MOD= 000004       CL$PLH= 034000       C.BUF   000014       EV$COZ= 000011       E$MOD   000012
AS.DEL= 000010       CL$PLL= 000600       C.BUF1  000014       EV$DBR= 000302       E$NOD   000010
AS.EXT= 000004       CL$PRT= 034200       C.BUF2  000024       EV$GAS= 035101       E$PORT  000014
AS.RED= 000001       CL$ROU= 010000       C.CNT   000020       EV$HCE= 035114       E$PRM   000002
AS.WRT= 000002       CL$SES= 000200       C.CNT1  000020       EV$HCI= 035113       E$STAT  000006
A$$CHK= 000000       CL$SGE= 035000       C.CNT2  000030       EV$HFE= 000506       E$TCB   000004
A$$CPS= 000000       CL$SSE= 035100       C.FLG   000022       EV$IFL= 000413       E$$XPR= 000000
A$$PRI= 000000       CL$TRN= 000400       C.FLG1  000022       EV$IFO= 000415       E.CTL   000020
A$$TRP= 000000       CL$XL2= 013700       C.FLG2  000032       EV$IFS= 000414       E.DATA  000046
A.IOC   000003       CL$XL3= 013600       C.FNC   000010       EV$INF= 000515       E.EVT   000002
A.LGTH= 000014       CL$X2S= 013500       C.LIN   000006       EV$LDL= 000407       E.LCN   000042
A.MPCT  000011       CM.CIR= 000002       C.LNK   000000       EV$LDN= 010416       E.LEN   000216
A.PCB   000012       CM.FMT= 100000       C.MOD   000011       EV$LDO= 000411       E.LIN   000024
A.PCBL  000000       CM.HRD= 000002       C.NSP   000004       EV$LDS= 000410       E.LNK   000000
A.PRI   000002       CM.LIN= 000000       C.PRO   000042       EV$LSC= 000500       E.MOD   000036
A.STAT  000010       CM.LOO= 000001       C.RSV   000002       EV$LUP= 000412       E.NOD   000034
A.TCB   000004       CM.XLO= 000004       C.STA   000007       EV$NOL= 000402       E.PDV   000021
A.TCBL  000006       CP.DCF= 000040       C.STS   000012       EV$NRC= 000416       E.PORT  000040
BLOCK   000152R  002 CP.HDL= 000007       C.URM   177776       EV$NSC= 000200       E.PRM   000026
BLXIO   000572RG 002 CP.PS = 177400       C.XACP  000004       EV$NUL= 000401       E.PVC   000044
CB.CCB= 000002       CP.PSI= 000200       C.XID   000035       EV$NVR= 000406       E.SIZ   000022
CB.DDM= 000040       CP.XCF= 000100       C.XLEN  000044       EV$OPL= 000403       E.TIME  000004
CB.DLC= 000020       CP.2FR= 000030       C.XPLI  000040       EV$PCC= 034000       FC.CCP= 000020
CB.RDB= 000004       CS.ABO= 000100       C.XPT   000034       EV$PCI= 034001       FC.CTL= 000006
CB.SDB= 000010       CS.BRO= 000002       C.XSVC  000042       EV$PCM= 034002       FC.KCP= 000016
CB.SLI= 000100       CS.BUF= 000200       C.XTC   000037       EV$PFE= 000404       FC.KIL= 000004
CB.XLB= 000001       CS.CES= 000002       C.X25   000036       EV$PPC= 034003       FC.MAN= 000024
CCBRT   000604RG 002 CS.CHN= 000010       DDMTBL  000164R  002 EV$RCF= 000517       FC.MLD= 000026
CC.LLC= 000200       CS.CMP= 000200       DEVERR  000051R  002 EV$RDC= 010001       FC.PCT= 000030
CE.ABO= 100362       CS.DCR= 000400       DL$AST= 000002       EV$RDR= 010002       FC.PWR= 000022
CE.DAO= 100346       CS.DEF= 000004       DL$HLT= 000000       EV$RJE= 035106       FC.RCE= 000002
CE.DIS= 100366       CS.DEV= 000002       DL$IST= 000001       EV$RSC= 000501       FC.RCP= 000014
CE.ERR= 100370       CS.DIS= 000040       DL$MAI= 000004       EV$RUL= 000405       FC.TIM= 000010
CE.ILN= 100350       CS.ENA= 000001       DL$OFF= 000001       EV$SNA= 035000       FC.XCP= 000012
CE.LTO= 100356       CS.ENB= 000020       DL$ON = 000000       EV$SNF= 000516       FC.XME= 000000
CE.MOP= 100372       CS.ERR= 100000       DL$RUN= 000003       EV$SPE= 035001       FR$BCC= 000007
CE.NTE= 100361       CS.FTL= 001000       DL$SHU= 000002       EV$XCE= 034110       FR$CCF= 000001
CE.RTE= 100376       CS.HCR= 000001       DL$SYN= 000005       EV$XDI= 013600       FR$CDF= 000002
CE.SRC= 100364       CS.HFE= 002000       DSPDRV  000414R  003 EV$XGW= 034111       FR$DAO= 000011
CE.STP= 100352       CS.LST= 040000       D$$BUG= 177514       EV$XMX= 000514       FR$EXC= 000000
CE.TME= 100354       CS.MTL= 004000       D$$ISK= 000000       EV$XRS= 000512       FR$FRM= 000010
CE.TMO= 100374       CS.RNG= 000010       D$$L11= 000001       EV$XSC= 000513       FR$FTL= 000005
CE.UNS= 100344       CS.ROV= 000004       D$$YNC= 000000       EV$X2S= 013500       FR$OPN= 000004
CF.CHN= 000001       CS.RSN= 010000       D$$YNM= 000000       EV.CCB= 000001       FR$RFD= 000006
CF.EOM= 000004       CS.SHU= 000001       EF$ACT= 000001       EV.CIR= 000020       FR$SBU= 000012
CF.HDR= 000020       CS.SID= 000002       ETYPCL= 010500       EV.LCB= 000100       FR$SHO= 000003
CF.LB = 100000       CS.STR= 000004       EVTBLK  000306R  002 EV.LIN= 000004       FR$UEJ  000013
CF.LIN= 000002       CS.SUC= 000001       EVTEFL= 000002       EV.MAP= 000002       FR$UPT= 000014
CF.SOM= 000010       CS.TMO= 020000       EVTLUN= 000002       EV.MOD= 000040       FS.AST= 000000
CF.SYN= 000040       CS.XMR= 000004       EV$ACF= 000201       EV.NOD= 000010       FS.CIB= 002000
CF.TRN= 000100       CTLNUM  000162R  002 EV$ADR= 000420       EV.PRT= 000200       FS.CRA= 001000
CL$ASZ= 010500       CURQST  000156R  002 EV$ADU= 000417       EXEVCL= 000017       FS.DIS= 013000
```

```
Symbol table
```

```
**FILE**ID**MANACP





MM        MM    AAAAAA    NN        NN    AAAAAA      CCCCCCCC  PPPPPPPP
MM        MM    AAAAAA    NN        NN    AAAAAA      CCCCCCCC  PPPPPPPP
MMMM    MMMM  AA      AA  NN        NN  AA      AA  CC          PP      PP
MMMM    MMMM  AA      AA  NN        NN  AA      AA  CC          PP      PP
MM  MM  MM    AA      AA  NNNN      NN  AA      AA  CC          PP      PP
MM  MM  MM    AA      AA  NNNN      NN  AA      AA  CC          PP      PP
MM        MM  AA      AA  NN  NN    NN  AA      AA  CC          PPPPPPPP
MM        MM  AA      AA  NN  NN    NN  AA      AA  CC          PPPPPPPP
MM        MM  AAAAAAAAAA  NN      NNNN  AAAAAAAAAA  CC          PP
MM        MM  AAAAAAAAAA  NN      NNNN  AAAAAAAAAA  CC          PP
MM        MM  AA      AA  NN        NN  AA      AA  CC          PP                ....
MM        MM  AA      AA  NN        NN  AA      AA  CC          PP                ....
MM        MM  AA      AA  NN        NN  AA      AA    CCCCCCCC  PP                ....
MM        MM  AA      AA  NN        NN  AA      AA    CCCCCCCC  PP                ....




LL              SSSSSSSS  TTTTTTTTTT
LL              SSSSSSSS  TTTTTTTTTT
LL            SS              TT
LL            SS              TT
LL            SS              TT
LL            SS              TT
LL              SSSSSS        TT
LL              SSSSSS        TT
LL                    SS      TT
LL                    SS      TT
LL                    SS      TT
LL                    SS      TT
LLLLLLLLLL    SSSSSSSS        TT
LLLLLLLLLL    SSSSSSSS        TT
```

MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28
Table of contents

```
                                                .SBTTL  MANINI - PROCESS NM.INI REQUESTS
                                        ;+
                                        ; **-MANINI-PROCESS NM.INI REQUESTS
                                        ;
                                        ; THIS ROUTINE PERFORMS ALLOCATES A CONTEXT AREA AND A WINDOW BLOCK,
                                        ; (UNLESS RF$NCX IS SET IN THE FUNCTION FLAGS WORD),AND CALLS THE
                                        ; CORRECT INITIAIZATION ROUTINE FOR THE REQUESTED FUNCTION.
                                        ;
                                        ; INPUTS:
                                        ;       R3 = ADDRESS OF I/O PACKET
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = I/O STATUS TO RETURN
                                        ;       R1 = NUMBER OF BYTES TO TRANSFER TO USER BUFFER
                                        ;       IF SUCCESS, CONTEXT ALLOCATED, AND WINDOW BLOCK CREATED.
                                        ;
                                        ; REGISTERS:
                                        ;       R3 IS PRESERVED
                                        ;-

002300                                  MANINI:
002300  005767  175716                          TST     TSKLN2                  ; IS THERE A WINDOW BLOCK ALREADY ?
002304  001353                                  BNE     ALNERR                  ; IF NE, YES - ERROR
002306  016705  175726                          MOV     MANTBL,R5               ; POINT TO MANAGEMENT FUNCTION TABLE
002312  016304  000024                          MOV     I.PRM(R3),R4            ; GET FUNCTIONS AND OPTIONS
002316  120427  000024                          CMPB    R4,#MF$REA              ; IS THIS A READ FUNCTION ?
002322  001403                                  BEQ     5$                      ; IF EQ, YES - CLEAR INFO TYPE
002324  120427  000224                          CMPB    R4,#MF$REA!CM$RSX       ; ELSE, IS IT A SYSTEM SPECIFIC READ ?
002330  001002                                  BNE     10$                     ; IF NE, NO
002332  042704  070000                  5$:     BIC     #<MO$INF*256.>,R4       ; CLEAR INFO TYPE FROM OPTION
002336  020415                          10$:    CMP     R4,(R5)                 ; IS THIS FUNCTION A MATCH ?
002340  001424                                  BEQ     20$                     ; IF EQ, YES
002342  062705  000010                          ADD     #R$SIZE,R5              ; ELSE, POINT TO NEXT FUNCTION ENTRY
002346  105765  000002                          TSTB    R$FLAG(R5)              ; AT END OF TABLE (RF$END SET) ?
002352  100371                                  BPL     10$                     ; IF PL, NO - LOOK AT NEXT ENTRY
002354  112700  177777                          MOVB    #ME.FUN,R0              ; ELSE, ILLEGAL FUNCTION OR OPTION
002360  012702  000001                          MOV     #1,R2                   ; LENGTH OF RESPONSE MESSAGE
002364  016363  000034  000024                  MOV     I.PRM+10(R3),I.PRM(R3)  ; REARRANGE BUFFER POINTER FOR
002372  016363  000036  000026                  MOV     I.PRM+12(R3),I.PRM+2(R3); ... COPYING DATA
002400  016363  000042  000030                  MOV     I.PRM+16(R3),I.PRM+4(R3); ... TO USER BUFFER
002406                                          CALLR   FINREQ                  ; PASS IT BACK TO USER AND RETURN

002412  132765  000001  000002  20$:    BITB    #RF$PRV,R$FLAG(R5)      ; IS THIS A PRIVILEGED REQUEST ?
002420  001414                                  BEQ     30$                     ; IF EQ, NO - SKIP PRIVILEGE CHECK
002422  016304  000004                          MOV     I.TCB(R3),R4            ; ELSE, GET USER'S TCB ADDRESS
002426  032764  000000G 177777G                 BIT     #T3.PRV,T.ST3-1(R4)     ; IS TASK PRIVILEGED ?
002434  001006                                  BNE     30$                     ; IF NE, YES - OK
002436  016404  177777G                         MOV     T.UCB-1(R4),R4          ; ELSE, GET UCB ADDRESS OF TASK'S TI:
002442  032764  000000G 000000G                 BIT     #U2.PRV,U.CW2(R4)       ; IS TERMINAL PRIVILEGED ?
002450  001705                                  BEQ     PRIERR                  ; IF EQ, NO - PRIVILEGE VIOLATION

002452                                  30$:    CALL    UPDCTX                  ; RE-WRITE THE LAST CONTEXT IF NEEDED
002456  132765  000002  000002                  BITB    #RF$NCX,R$FLAG(R5)      ; DOES THIS FUNCTION NEED A CONTEXT ?
002464  001021                                  BNE     40$                     ; IF NE, NO - SKIP ALLOCATION
002466                                          CALL    ALLCTX                  ; ALLOCATE CONTEXT AREA
002472  103666                                  BCS     NDRERR                  ; IF CS, INSUFFICIENT RESOURCES
002474  122763  002401  000012                  CMPB    #NM.VR2,I.FCN(R3)       ; IS THIS REQUEST FROM V2.0 NCP?
```

```
                                                                        C  5
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 3
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE             REFERENCES

LSTCTX     000214 RG      #5-262     *6-438
MANACP     001472 RG      #6-302      11-696
MANCLN     0027[illegible] R        6-474    #10-646
MANINI     0023[illegible] R        6-472    #8-537
MANOPR     0026[illegible] R        6-473    #9-615
MANTBL     000240 R       #5-272      8-540
MAXID    = 000062         #5-78       5-277       6-400
MAXLID   = 000020         #5-79       5-204
ME.DON     177600           9-624
ME.FUN     177777           8-552
MF$REA     000024           8-542     8-544
MN$VOL   = 000000         #5-74       6-318
MOULS      000034 RG      #5-98
MO$ENT     000017           8-581
MO$INF     000160           8-546
NDRERR     002250 R       #7-503      8-572
NLNERR     002256 R       #7-506      9-617
NMCLH      000120 RG      #5-124      6-326
NMCL2      000122 RG      #5-132      6-336
NMCRS      000124 RG      #5-136
NMLST      000126 RG      #5-137
NMVVCL   = 000102  G      #5-180
NMVVEC     000000 RG      #5-83       5-180
NM.CLN     003400           6-474
NM.INI     002400           6-395     6-472
NM.OPR     003000           6-473
NM.VR2     002401           8-573
NM.VR3     002402           8-576
NTLPT      000130 RG      #5-138
NUMUSR     000220 RG      #5-264      6-363
OBJHD      000132 RG      #5-139
PDVID      000134 RG      #5-140
PDVNM      000136 RG      #5-141
PDVTA      000140 RG      #5-142
POOLPT     000234 RG      #5-270
POOLSZ     000236 RG      #5-271
PRIERR     002264 R       #7-509      8-566
PSIPT      000142 RG      #5-143
RBUFF    = ******  GX       6-399
RDBAF      000144 RG      #5-144
RDBCT      000146 RG      #5-145
RDBNM      000150 RG      #5-146
RDBSZ      000152 RG      #5-147
RDBTH      000154 RG      #5-148
RELOC      000036 RG      #5-99
RF$END   = 000200  G      #5-214
RF$NCX   = 000002  G      #5-216      8-569
RF$PRV   = 000001  G      #5-215      8-559
RID      = ******  GX       6-404
R$FLAC     000002  G      #5-213      8-550       8-559       8-569
R$FUNC     000000  G      #5-212
R$INIT     000004  G      #5-218      8-591
```

```
                                                                        D  5
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 4
```

```
MANAST     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 2

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME          REFERENCES

ASTX$S             #5-57        6-92
CALL                6-88
CALLR               6-90
DIR$               #6-92        6-92
SWSTK$              6-88
```

```
MAN
Sym

AAL
AB.
AB.
AF.
AF.
AF.
AF.
AF.
AF.
AF.
AF.
AK.
AK.
AK.
AK.
ALL
AS.
AS.
AS.
AS.
AS.
AS.
AS.
AS.
A$$
A$$
A$$
A$$
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
A.
```

```
**FILE**ID**MANBUF
```

```
MAN
Sym
```

```
AALOCB= ****** GX       A.RES = 000032          CP.DSB= 000010          C.BUF2  000024          FS.DIS= 013000
AB.NPV= 000001          A.SBUF  000022          CP.EXT= 000400          C.CNT   000020          FS.DVC= 001000
AB.TYP= 000002          A.SLEN  000024          CP.HDL= 000007          C.CNT1  000020          FS.ENB= 012000
AF.AST= 000010          A.SMAP  000020          CP.LGO= 000004          C.CNT2  000030          FS.EXI= 001000
AF.ESQ= 000020          A.STA   000015          CP.MSG= 000002          C.FLG   000022          FS.GET= 006000
AF.LCK= 000040          A.TRAN  000022          CP.NIO= 000100          C.FLG1  000022          FS.HLT= 000000
AF.MDE= 000200          BLXIO = ****** GX       CP.NUL= 000001          C.FLG2  000032          FS.INI= 000000
AF.NOT= 000002          BUFFER= ****** GX       CP.PRV= 000020          C.FNC   000010          FS.KIL= 000000
AF.OOB= 000004          BUFTBL  000306R     002 CP.PS = 177400          C.LIN   000006          FS.LCL= 100000
AF.QUE= 000100          CB.CCB= 000002          CP.PSI= 000200          C.LNK   000000          FS.LTM= 001000
AF.XCC= 000001          CB.DDM= 000040          CP.RST= 000200          C.MOD   000011          FS.MNT= 004000
AK.BUF= 000200          CB.DLC= 000020          CP.SGL= 000040          C.NSP   000004          FS.MSN= 014000
AK.GBI= 000202          CB.RDB= 000004          CP.XCF= 000100          C.PCPL  000011          FS.REA= 001000
AK.GGF= 000303          CB.SDB= 000010          CP.2FR= 000030          C.PDPL  000010          FS.RET= 000000
AK.OCB= 000201          CB.SLI= 000100          CSBRT = ****** GX       C.PNAM  000002          FS.REZ= 003000
ALLCTX  000000RG        CB.XLB= 000001          CS.ABO= 000100          C.PRMT  000012          FS.RLB= 002000
AS.CAA= 000006          CCBRT = ****** GX       CS.BRO= 000002          C.PRO   000042          FS.RNG= 011000
AS.DIS= 000002          CC.LLC= 000200          CS.BUF= 000200          C.PSTS  000006          FS.RST= 000000
AS.DLT= 000001          CE.ABO= 100362          CS.CES= 000002          C.PTCB  000000          FS.RTN= 001000
AS.FPA= 000001          CE.DAO= 100346          CS.CHN= 000010          C.RSV   000002          FS.SET= 005000
AS.PFA= 000004          CE.DIS= 100366          CS.CMP= 000200          C.STA   000007          FS.SFC= 005000
AS.RCA= 000002          CE.ERR= 100370          CS.DCR= 000400          C.STS   000012          FS.SFR= 006000
AS.REA= 000005          CE.ILN= 100350          CS.DEF= 000004          C.URM   177776          FS.SFS= 004000
AS.RRA= 000003          CE.LTO= 100356          CS.DEV= 000002          C.XACP  000004          FS.SPW= 040000
A$$CHK= 000000          CE.MOP= 100372          CS.DIS= 000040          C.XID   000035          FS.STM= 000000
A$$CPS= 000000          CE.NTE= 100361          CS.ENA= 000001          C.XLEN  000044          FS.STP= 002000
A$$PRI= 000000          CE.RTE= 100376          CS.ENB= 000020          C.XPLI  000040          FS.STR= 001000
A$$TRP= 000000          CE.SRC= 100364          CS.ERR= 100000          C.XPT   000034          FS.TRM= 003000
A.ACC   000014          CE.STP= 100352          CS.FTL= 001000          C.XSVC  000042          FS.WLB= 001000
A.ACCE  000000          CE.TME= 100354          CS.HCR= 000001          C.XTC   000037          FS.XKL= 002000
A.AST   000006          CE.TMO= 100374          CS.HFE= 002000          C.X25   000036          FS.XOF= 010000
A.BYT   000004          CE.UNS= 100344          CS.LST= 040000          DDEACB= ****** GX       FS.XON= 007000
A.CALL  000024          CF.CHN= 000001          CS.MTL= 004000          DEACTX  000264RG        FS.ZER= 002000
A.CBL   000002          CF.EOM= 000004          CS.RNG= 000010          D$$BUG= 177514          F$$LVL= 000001
A.CONN  000012          CF.HDR= 000020          CS.ROV= 000004          D$$ISK= 000000          GS.DEL= 000001
A.DEQU  000002          CF.LB = 100000          CS.RSN= 010000          D$$L11= 000001          G$$TPP= 000000
A.DIS   000002          CF.LIN= 000002          CS.SHU= 000001          D$$YNC= 000000          G$$TSS= 000000
A.DISC  000014          CF.SOM= 000010          CS.SID= 000002          D$$YNM= 000000          G$$TTK= 000000
A.DQSR  177776          CF.SYN= 000040          CS.STR= 000004          E$$XPR= 000000          G$$WRD= 000000
A.FLEN  000011          CF.TRN= 000100          CS.SUC= 000001          FC.CCP= 000020          G.CNT   000004
A.IBUF  000014          CM.CDS= 000004          CS.TMO= 020000          FC.CTL= 000006          G.EFLG  000006
A.ILEN  000016          CM.CEN= 000003          CS.XUR= 000004          FC.KCP= 000016          G.GRP   000002
A.IMAP  000012          CM.CIR= 000002          CTXHD = ****** GX       FC.KIL= 000004          G.LGTH= 000012
A.INPU  000006          CM.ELM= 000005          CTXP00  000000R     002 FC.MAN= 000024          G.LNK   000000
A.IOS   000026          CM.EXT= 000006          CTXSIZ= ****** GX       FC.MLD= 000026          G.STAT  000003
A.KSR5  177774          CM.FMT= 100000          CURCTX= ****** GX       FC.PCT= 000030          G.TSDU= 000036
A.LIN   000012          CM.HRD= 000002          CURIOP= ****** GX       FC.PWR= 000022          G.TSFW= 000024
A.MAS   000004          CM.IND= 000002          C$DATA= ****** GX       FC.RCE= 000002          G.TSGC= 000017
A.NPR   000010          CM.INE= 000001          C$FLAG= ****** GX       FC.RCP= 000014          G.TSMT= 000022
A.NUM   000010          CM.LIN= 000000          C$SIZE= ****** GX       FC.TIM= 000010          G.TSNL= 000020
A.OUTP  000010          CM.LKT= 000007          C$$CKP= 000000          FC.XCP= 000012          G.TSPC= 000016
A.PLEN= 000016          CM.LOO= 000001          C$$ORE= 000400          FC.XME= 000000          G.TSPN= 000004
A.POWE  000004          CM.MSG= 000011          C$$RSH= 177564          FE.CEX= ****** GX       G.TSPR= 000014
A.PRM   000012          CM.RMT= 000010          C.ADD   000034          FMASK = ****** GX       G.TSRN= 000010
A.PROC  000020          CM.XLO= 000004          C.BID   000003          FS.AST= 000000          G.TSSY= 000034
A.RECE  000016          CONTEX= ****** GX       C.BUF   000014          FS.CIB= 002000          G.TSTN= 000000
A.REL   000000          CP.DCF= 000040          C.BUF1  000014          FS.CRA= 001000          G.TSTS= 000032
```

```
     86                                                     .SBTTL  LOGOUT - PERFORM LOGOUT CLEANUP FOR A USER
     87                                             ;+
     88                                             ; **-LOGOUT-PERFORM LOGOUT CLEANUP FOR A USER
     89                                             ;
     90                                             ; THIS ROUTINE IS CALLED TO DELETE ALL LOCAL ALIASES FOR A USER.
     91                                             ;
     92                                             ; INPUTS:
     93                                             ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
     94                                             ;       R3 = ADDRESS OF INPUT/OUTPUT BUFFER
     95                                             ;       R4 = ADDRESS OF CONTEXT AREA
     96                                             ;
     97                                             ; OUTPUTS:
     98                                             ;       R0 = NICE RETURN CODE (MS.SUC)
     99                                             ;       R2 = LENGTH OF RETURN DATA (0)
    100                                             ;       R3 = ADDRESS OF OUTPUT BUFFER
    101                                             ;
    102                                             ; REGISTERS:
    103                                             ;       R3, R4, R5 ARE PRESERVED
    104                                             ;-
    105
    106 000000                                  LOGOUT::
    107 000000  012700  000000'                         MOV     #BYETBL,R0                      ; POINT TO START OF TABLE
    108 000004  005720                          10$:    TST     (R0)+                           ; AT END OF TABLE ?
    109 000006  001410                                  BEQ     20$                             ; IF EQ, YES - ALL DONE
    110 000010                                          SAVRG   <R0,R2,R3>                      ; ELSE, SAVE A FEW REGISTERS
    111 000016                                          CALL    @-(R0)                          ; CALL THE ROUTINE
    112 000020                                          RESRG   <R3,R2,R0>                      ; RESTORE WHAT WE SAVED
    113 000026  000766                                  BR      10$                             ; ... AND SO ON
    114
    115 000030  005002                          20$:    CLR     R2                              ; DON'T RETURN NOTHING!
    116 000032  012700  000001                          MOV     #MS.SUC,R0                      ; SET SUCCESSFUL RETURN
    117 000036                                          RETURN
```

```
000332                          50$:    .BLKW   1               ; ... WITH ADDRESS OF SEG DESCRIPTOR
000334  000316'                         .WORD   40$             ; ... AND TRANSFER ADDRESS
                                ;
                                ; RESTORE ''BUFFER'' FROM THE USER'S BUFFER
                                ;
000336                          60$:    SWSTK$  70$             ;; ENTER SYSTEM STATE
000342  016703  000000G                 MOV     CURIOP,R3       ;; GET I/O PACKET ADDRESS
000346  062703  000024                  ADD     #I.PRM,R3       ;; POINT TO BUFFER DESCRIPTOR
000352                                  CALLR   READ            ;; ''READ'' THE USER'S BUFFER BACK IN
000356                          70$:    RETURN                  ; RESTORE REGS AND RETURN VIA SAVAL

        000001                          .END
```

```
                                .SBTTL  PRIVILEGED ROUTINE ENTRY POINTS

                                .ENABL  LSB
                        ;
                        ; ALIAS INITIALIZE
                        ;
000420  012704          ALCHIN::MOV     (PC)+,R4
000422     010                  .BYTE   ME$ALI
000423     223                  .BYTE   MF$CHA!CM$RSX
000424  000435                  BR      ALID
000426  012704          ALREIN::MOV     (PC)+,R4
000430     010                  .BYTE   ME$ALI
000431     224                  .BYTE   MF$REA!CM$RSX
000432  000432                  BR      ALID
                        ;
                        ; AREA INITIALIZE
                        ;
000434  012704          ARREIN::MOV     (PC)+,R4
000436     005                  .BYTE   ME$ARE
000437     024                  .BYTE   MF$REA
000440  000504                  BR      ARID
                        ;
                        ; CIRCUIT INITIALIZE
                        ;
000442  012704          CICHIN::MOV     (PC)+,R4
000444     003                  .BYTE   ME$CIR
000445     023                  .BYTE   MF$CHA
000446  000424                  BR      CIID
000450  012704          CIREIN::MOV     (PC)+,R4
000452     003                  .BYTE   ME$CIR
000453     024                  .BYTE   MF$REA
000454  000421                  BR      CIID
000456  012704          CIZEIN::MOV     (PC)+,R4
000460     003                  .BYTE   ME$CIR
000461     025                  .BYTE   MF$ZER
000462  000416                  BR      CIID
                        ;
                        ; LINE INITIALIZE
                        ;
000464  012704          LICHIN::MOV     (PC)+,R4
000466     001                  .BYTE   ME$LIN
000467     023                  .BYTE   MF$CHA
000470  000413                  BR      LIID
000472  012704          LIREIN::MOV     (PC)+,R4
000474     001                  .BYTE   ME$LIN
000475     024                  .BYTE   MF$REA
000476  000410                  BR      LIID
000500  012704          LIZEIN::MOV     (PC)+,R4
000502     001                  .BYTE   ME$LIN
000503     025                  .BYTE   MF$ZER
000504  000405                  BR      LIID
                        ;
                        ; PROCESS INITIALIZE
                        ;
000506  012704          PRCHIN::MOV     (PC)+,R4
000510     005                  .BYTE   ME$PRO
000511     223                  .BYTE   MF$CHA!CM$RSX
```

Table of contents

```
     5-   50     MACRO CALLS AND LOCAL DATA
     6-   65     TRAADD - TRANSLATE AN ADDRESS TO A NAME
     7-   98     TRANAM - TRANSLATE A NAME TO AN ADDRESS
     8-  131     TRAALI - TRANSLATE AN ALIAS TO AN NAME
     9-  166     TRAACC - TRANSLATE AN ALIAS TO AN NAME AND ACCESS CONTROL
```

```
**FILE**ID**MANVCD
```

```
MM      MM    AAAAAA    NN      NN  VV      VV    CCCCCCCC  DDDDDDDD
MM      MM    AAAAAA    NN      NN  VV      VV    CCCCCCCC  DDDDDDDD
MMMM  MMMM  AA      AA  NN      NN  VV      VV  CC          DD      DD
MMMM  MMMM  AA      AA  NN      NN  VV      VV  CC          DD      DD
MM  MM  MM  AA      AA  NNNN    NN  VV      VV  CC          DD      DD
MM  MM  MM  AA      AA  NNNN    NN  VV      VV  CC          DD      DD
MM      MM  AA      AA  NN  NN  NN  VV      VV  CC          DD      DD
MM      MM  AA      AA  NN  NN  NN  VV      VV  CC          DD      DD
MM      MM  AAAAAAAAAA  NN    NNNN  VV      VV  CC          DD      DD
MM      MM  AAAAAAAAAA  NN    NNNN  VV      VV  CC          DD      DD
MM      MM  AA      AA  NN      NN    VV  VV    CC          DD      DD      ....
MM      MM  AA      AA  NN      NN    VV  VV    CC          DD      DD      ....
MM      MM  AA      AA  NN      NN      VV        CCCCCCCC  DDDDDDDD        ....
MM      MM  AA      AA  NN      NN      VV        CCCCCCCC  DDDDDDDD        ....
```

```
LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
                .TITLE  MANVTB - VOLATILE MANAGEMENT ACP TABLES
                .IDENT  /V05.00/
        ;
        ; COPYRIGHT (C) 1982, 1983, 1985 BY
        ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
        ;
        ;
        ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
        ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
        ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
        ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
        ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
        ; TRANSFERRED.
        ;
        ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
        ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
        ; CORPORATION.
        ;
        ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
        ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
        ;
        ;
        ; MODULE DESCRIPTION:
        ;
        ;       MANVTB - TABLES SPECIFIC TO THE VOLATILE MANAGEMENT ACP
        ;
        ;
        ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
        ;
        ; IDENT HISTORY:
        ;
        ; 1.00  16-APR-82
        ;       DECNET-11M V3.1
        ;       DECNET-11M-PLUS V1.1
        ;
        ; 4.00  07-NOV-83
        ;       DECNET-11M V4.0
        ;       DECNET-11M-PLUS V2.0
        ;
        ; 5.00  22-JUL-85
        ;       DECnet-11M/S V4.2
        ;       DECnet-11M-Plus V3.0
        ;       DECnet-Micro/RSX V1.0
        ;
        ;
        .TITLE  PSIBYE - Perform logout cleanup for PSI
        .IDENT  /V05.00/
        .ENABL  LC
        .NLIST  BEX
;
; COPYRIGHT (C) 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; Deallocate per-terminal PSI data structures when a user logs out
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  29-May-83
;       First written
;       [Tim Dixon EDSSEG]
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
;
```

```
                                        .SBTTL  CHKDES - CHECK DESTINATION FORMAT
                                ;+
                                ; **-CHKDEC-CHECK DESTINATION PARAMETER FORMAT
                                ;
                                ; THIS ROUTINE CHECKS THE PASSED DESTINATION PARAMETER FORMAT FOR
                                ; VALIDITY, AND IF VALID, SAVES THE DESTINATION IN THE CONTEXT AREA.
                                ;
                                ; INPUTS:
                                ;       R2 = NUMBER OF BYTES REMAINING IN INPUT BUFFER
                                ;       R3 = ADDRESS OF REMAINING INPUT
                                ;       R4 = ADDRESS OF CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       R2 = NUMBER OF BYTES REMAINING IN INPUT BUFFER
                                ;       R3 = ADDRESS OF REMAINING INPUT
                                ;       C-BIT ERROR INDICATOR
                                ;         IF CLEAR, A$DES = SAVED DESTINATION INFORMATION, A$NUM = LENGTH OF
                                ;            ALIAS BLOCK TO ALLOCATE
                                ;         IF SET, ERROR RETURN IN R0, PARAMETER TYPE SET IN A$PAR
                                ;
                                ; REGISTERS:
                                ;       R4 IS PRESERVED
                                ;-

000272                          CHKDES:
                                ;
                                ; CHECK PARAMETER TYPE = DESTINATION
                                ;
000272  005064  000004                  CLR     A$PAR(R4)               ; NO ''CURRENT PARAMETER'' YET
000276  012701  000156                  MOV     #MP$DES,R1              ; SET PARAMETER TYPE TO CHECK FOR
000302                                  CALL    CHKPAR                  ; CHECK FOR THE PARAMETER
000306  103467                          BCS     40$                     ; IF CS, ERROR - RETURN
000310  110164  000004                  MOVB    R1,A$PAR(R4)            ; CURRENT PARAMETER = DESTINATION
                                ;
                                ; CHECK AND SAVE THE DESTINATION NAME
                                ;
000314  010400                          MOV     R4,R0                   ; COPY THE CONTEXT BLOCK POINTER
000316  062700  000022                  ADD     #A$DES,R0               ; POINT TO THE DESTINATION NAME AREA
000322  112305                          MOVB    (R3)+,R5                ; GET NAME BYTE COUNT, SIGN EXTENDED
000324  003014                          BGT     10$                     ; IF GT, SPECIFIED BY NAME
000326  002454                          BLT     30$                     ; IF LT, ILLEGAL FORMAT
000330  162702  000003                  SUB     #3,R2                   ; SUB DESTINATION FIELD LENGTH
000334  005001                          CLR     R1                      ; ELSE, INIT NODE ADDRESS
000336  152301                          BISB    (R3)+,R1                ; GET LOW BYTE OF ADDRESS
000340  000301                          SWAB    R1                      ; ...
000342  152301                          BISB    (R3)+,R1                ; GET HIGH BYTE
000344  000301                          SWAB    R1                      ; ...
000346                                  CALL    MAPADD                  ; MAP ADDRESS INTO NAME
000352  103442                          BCS     30$                     ; IF CS, NO NAME FOUND - ERROR
000354  000411                          BR      20$                     ; ELSE, CONTINUE WITH ACCESS

000356  005303                  10$:    DEC     R3                      ; POINT TO BEGINNING OF BUFFER
000360  010301                          MOV     R3,R1                   ; COPY BUFFER ADDRESS
000362  005302                          DEC     R2                      ; DEC BUFF COUNT BY NAME LENGTH
000364  160502                          SUB     R5,R2                   ; ...
000366  002434                          BLT     30$                     ; IF LT, BUFFER TOO SMALL - ERROR
000370                                  CALL    SAVNAM                  ; CHECK A NODE NAME  AND SAVE IT
```

```
                                        .SBTTL  SETALI - SET AN ALIAS
                                ;+
                                ; **-SETALI-SET AN ALIAS
                                ;
                                ; THIS ROUTINE IS CALLED TO BUILD AN ALIAS NAME BLOCK (ANB) AND LINK
                                ; IT INTO THE ALIAS LIST OFF OF D$ANN. IF AN ALIAS ALREADY EXISTS
                                ; WITH THE SAME NAME AND SCOPE, IT IS DELETED BEFORE THE NEW ONE IS
                                ; BUILT.
                                ; THE ALIAS LIST IS ORDERED BY SCOPE (NUMERIC UCB/PUD ADDRESS), WITH THE
                                ; EXCEPTION OF GLOBAL ALIASES WHICH ARE AT THE END OF THE LIST.
                                ;
                                ; INPUTS:
                                ;       R4 = ADDRESS OF CONTEXT AREA WITH THE FOLLOWING OFFSETS FILLED IN:
                                ;               A$NUM = LENGTH OF ANB TO ALLOCATE
                                ;               A$ALI = ALIAS NAME, SPACE FILLED
                                ;               A$SCO = SCOPE (0=GLOBAL, ELSE UCB ADDRESS OF TERMINAL)
                                ;               A$DES = DESTINATION NAME, SPACE FILLED, FOLLOWED BY
                                ;                       ACCESS CONTROL FIELDS
                                ;
                                ; OUTPUTS:
                                ;       R0 = RETURN CODE
                                ;
                                ; REGISTERS:
                                ;       R4 IS PRESERVED
                                ;-

000072                          SETALI:
000072  012700  000001                  MOV     #MS.SUC,R0              ; ASSUME SUCCESS

                                        .IF DF  R$$PRO                  ; PRO/DECNET

                                        MOV     #DF.AUP,R1              ; SET ALIAS UPDATE BIT
                                        CALL    CKLOCK                  ; WAIT UNTIL NT.CFG PARTITION IS UNLOCKED
                                        BCC     5$                      ; IF CC, UNLOCKED
                                        MOV     #ME.OPE,R0              ; ELSE OPERATION FAILURE
                                        RETURN

                                        .ENDC   ; IF DF R$$PRO

000076                          5$:     SWSTK$  100$                    ;; ENTER SYSTEM STATE
000102                                  SAVMAP                          ;; SAVE CURRENT MAPPING
000106  017700  000000G                 MOV     @DECPT,R0               ;; POINT TO DEC HOME BLOCK
000112  001506                          BEQ     90$                     ;; NO HOME BLOCK, NO ALIASES
000114  062700  000000                  ADD     #D$ANN,R0               ;; POINT TO ALIAS NAME LIST
                                ;
                                ; LOOK FOR AN EXISTING ALIAS, IF FOUND, DELETE IT
                                ;
000120                          10$:    CALL    FINDA                   ;; LOOK FOR MATCHING ALIAS NAME
000124  103412                          BCS     30$                     ;; IF CS, NAME IS NOT A DUPLICATE
000126  026164  000010  000020          CMP     A.UCB(R1),A$SCO(R4)     ;; ELSE, DOES IT HAVE THE SAME OWNER ?
000134  001003                          BNE     20$                     ;; IF NE, NO - CONTINUE SCAN
000136                                  CALL    DEAANB                  ;; DEALLOCATE OLD ALIAS NAME BLOCK
000142  000403                          BR      30$                     ;; AND CREATE NEW ONE
000144  016700  000000G         20$:    MOV     $CRUNM,R0               ;; CONTINUE SCAN FROM THIS BLOCK
000150  000763                          BR      10$                     ;; ...
                                ;
                                ; ALOCATE NEW ALIAS BLOCK
```

```
000470  016704  000150'                   MOV     LDBADR,R4               ;; Get APR6-biased LDB address
000474                                    CALL    @BLXIO                  ;; Copy microcode to LDB
000500  016704  000144'                   MOV     LDBPTR,R4               ;; Get address of LDB's CCB
000504  010564  000020                    MOV     R5,C.CNT1(R4)           ;; Copy LDB byte count to CCB
000510  016764  000150' 000016            MOV     LDBADR,C.BUF1+2(R4)     ;; Recopy address if driver changed it

                                          .ENDC                           ;; End UMR conditional



                                  ;       Pick up the information needed to dispatch to the driver and perform
                                  ;       the dispatch.  Return to user state after the dispatch - all registers
                                  ;       and mappings prior to the switch into system state are preserved across
                                  ;       the transition.


000516  016702  000160'           10$:    MOV     PDVIDX,R2               ;; Get driver's PDV index
000522  016705  000164'                   MOV     DDMTBL,R5               ;; Get line table address as well
000526  010403                            MOV     R4,R3                   ;; Copy CCB address
000530  062703  000010                    ADD     #C.FNC,R3               ;; Compute address of function code
000534                                    CALLR   @PDSPL                  ;; Dispatch to driver at process level
                                                                          ;;     and return to user state at "20$"

000540  105764  000011            20$:    TSTB    C.MOD(R4)               ; Clear C-bit and test for halt
000544  001426                            BEQ     40$                     ; Branch if "halt" CCB, ignore status

                                          .IF     DF      N$$UMR          ; If system has UMRs ...

000546  016705  000144'                   MOV     LDBPTR,R5               ; Get address of LDB's CCB
000552  016503  000012                    MOV     C.STS(R5),R3            ; Get driver's load status
000556  001011                            BNE     25$                     ; Branch if error(s) occurred
000560  067764  000616' 000016            ADD     @RDBSZ,C.BUF1+2(R4)     ; Advance load buffer address
000566  167764  000616' 000020            SUB     @RDBSZ,C.CNT1(R4)       ; Subtract enough data to fill an LDB
000574  003307                            BGT     DSPDRV                  ; Branch if more of buffer to load
000576  000241                            CLC                             ; Else clear C-bit to continue load
000600                                    RETURN                          ; Exit

000602                            25$:    .IFF                            ; If system has no UMRs ...

                                          MOV     C.STS(R4),R3            ; Else, get driver's completion status
                                          BEQ     40$                     ; Branch if loading is to continue

                                          .ENDC                           ; End UMR conditional

000602  032703  077777                    BIT     #^C<CS.ERR>,R3          ; Errors other than logical data end?
000606  001404                            BEQ     30$                     ; Branch if not
000610  012702  000051'                   MOV     #DEVERR,R2              ; Get "device error" string address
000614                                    CALL    LOGEVT                  ; Go log the event

000620  000261                    30$:    SEC                             ; Set C-bit to indicate load halt
000622                            40$:    RETURN                          ; Exit
```

```
FS.DVC= 001000        LF.LPB= 001000        M$$CRB= 000124        P.IOC   000003        SF.MFL= 000040
FS.ENB= 012000        LF.MDC= 000100        M$$CRX= 000000        P.LNK   000000        SF.PAC= 000020
FS.EXI= 001000        LF.MFL= 004000        M$$FCS= 000000        P.MAIN  000012        SF.REA= 000010
FS.GET= 006000        LF.MTP= 000020        M$$MGE= 000000        P.NAM   000004        SF.SER= 000001
FS.HLT= 000000        LF.PAC= 000200        M$$NET= 000000        P.OWN   000026        SF.SVC= 000002
FS.INI= 000000        LF.RDY= 040000        M$$OVR= 000000        P.PRI   000002        SF.UNL= 000040
FS.KIL= 000000        LF.REA= 010000        M$3100= 000000        P.REL   000014        SLOAD   001330R   003
FS.LCL= 100000        LF.SER= 000040        M$3101= 000001        P.SIZE  000016        SLTMA   000620RG  002
FS.LTM= 001000        LF.TIM= 000010        M$3102= 000002        P.STAT  000030        SRNAM   001246R   003
FS.MNT= 004000        LF.UNL= 020000        M$3103= 000003        P.SUB   000010        STDDM   000622RG  002
FS.MSN= 014000        LF.X2P= 000000        M.KTAE= 000010        P.SWSZ  000022        SV$DUM= 000001
FS.REA= 001000        LN.CLO= 000000        M.KTEF= 000002        P.TCB   000026        SV$LDA= 000000
FS.RET= 000000        LN.DUM= 000005        M.KTMG= 000004        P.WAIT  000020        S$$WRG= 000000
FS.REZ= 003000        LN.LOA= 000004        M.KTUN= 000006        QRMVF   000600RG  002 S$$YSZ= 007600
FS.RLB= 002000        LN.LOO= 000003        NBUFS = 000001        Q$$OPT= 000010        S.COST  000001
FS.RNG= 011000        LN.OAU= 000003        NTUIC   000574RG  002 RDBSZ   000616RG  002 S.FLG   000000
FS.RST= 000000        LN.OFF= 000001        N$$ACC= 000001        RE$ADC= 000004        S.LEN   000004
FS.RTN= 001000        LN.ON = 000000        N$$BUF= 000001        RE$ADF= 000017        S.NMST  000002
FS.SET= 005000        LN.OOP= 000004        N$$LDV= 000001        RE$ADR= 000007        S.OWNR  000003
FS.SFC= 005000        LN.OPE= 000001        N$$MCP= 000001        RE$BLK= 000010        S.TSEF= 000002
FS.SFR= 006000        LN.REF= 000002        N$$MLL= 000001        RE$CAF= 000014        TKTCB   000602RG  002
FS.SFS= 004000        LN.SER= 000002        N$$MOV= 000010        RE$DAT= 000001        TMREFL= 000003
FS.SPW= 040000        LN.STA= 000017        N$$NCT= 000001        RE$DRP= 000016        TS.BLK= 171700
FS.STM= 000000        LN.SUB= 000360        N$$PEM= 000001        RE$LDT= 000013        TS.CKP= 000200
FS.STP= 002000        LN.TRI= 000006        N$$UMR= 000001        RE$LSN= 000012        TS.CKR= 000100
FS.STR= 001000        LOCPAR  001074R   003 OP$INI= 000000        RE$NML= 000001        TS.EXE= 100000
FS.TRM= 003000        LOGEVT  000624R   003 OP$TER= 000001        RE$OPE= 000004        TS.HLD= 002000
FS.WLB= 001000        L$$ASG= 000000        PARERR  000073R   002 RE$OPR= 000000        TS.MSG= 020000
FS.XKL= 002000        L$$DRV= 000000        PARHD   000576RG  002 RE$RCV= 000001        TS.NRP= 010000
FS.XDF= 010000        L$$P11= 000001        PCBADR  000124R   002 RE$SED= 000011        TS.OUT= 000400
FS.XON= 007000        L$$11R= 000000        PDSPL   000612RG  002 RE$SKW= 000006        TS.RDN= 040000
FS.ZER= 002000        L.COST  000015        PDVIDX  000160R   002 RE$STA= 000002        TS.RUN= 004000
F$$LVL= 000001        L.CTL   000012        PDVTA   000614RG  002 RE$SUM= 000003        TS.STP= 001000
G$$TPP= 000000        L.CVA   177776        PH$HDE= 000004        RE$SYN= 000000        T$$KMG= 000000
G$$TSS= 000000        L.DDM   000002        PH$LOC= 000002        RE$TME= 000021        T$$MIN= 000000
G$$TTK= 000000        L.DDS   000004        PH$MTS= 000003        RE$TMO= 000000        T.ACTL  000052
G$$WRD= 000000        L.DLC   000003        PH$UMP= 000000        RE$TMR= 000020        T.ASTL  000016
IO.WLB= ****** GX     L.DLM   000006        PH$WCS= 000001        RE$UPT= 000002        T.ATT   000062
ISAR5   000570RG  002 L.DLS   000010        PS.APR= 000007        RE$URE= 000003        T.CPCB  000004
I$$RAR= 000000        L.FLG   000000        PS.CHK= 010000        RE$VER= 000005        T.DPRI  000040
I$$RDN= 000000        L.KRBA  000016        PS.CKP= 040000        RE$VRQ= 000015        T.EFLG  000022
KISAR5= ****** GX     L.LEN = 000022        PS.CKR= 020000        RT$INI= 000002        T.IOC   000003
K$$CNT= 177546        L.MPF   000022        PS.COM= 000200        RT$OFF= 000001        T.LBN   000041
K$$CSR= 177546        L.NMST  000020        PS.DEL= 000010        RT$ON = 000000        T.LDV   000044
K$$LDC= 000000        L.NSTA  000014        PS.DRV= 000020        R$$DER= 000000        T.LNK   000000
K$$TPS= 000074        L.OWNR  000021        PS.FXD= 004000        R$$K11= 000001        T.MXSZ  000050
LDBADR  000150R   002 L.UNT   000013        PS.LIO= 001000        R$$SND= 000000        T.NAM   000006
LDBGT   000606RG  002 MLD     000000R   003 PS.NSF= 000400        R$$11M= 000000        T.OFF   000066
LDBIAS  000146R   002 MLDERR  000000R   002 PS.OUT= 100000        R$$11S= 000001        T.PCB   000046
LDBPTR  000144R   002 MO$SAC= 000016        PS.PER= 002000        SC$OFF= 000001        T.PRI   000002
LDBRT   000610RG  002 MO$SPR= 000012        PS.PIC= 000100        SC$ON = 000000        T.RCVL  000012
LDBWTS= 000001        MO$SSV= 000014        PS.SYS= 000040        SC$RST= 000003        T.RRFL  000072
LD$LP = 000000        MO$25A= 000006        PWRAST  000754R   003 SC$SHU= 000002        T.SAST  000054
LF.ACT= 100000        MO$25P= 000002        P$$P45= 000000        SETTMR  000126R   002 T.SRCT  000071
LF.BRO= 000400        MO$25S= 000004        P$$WRD= 000000        SF.ACT= 000200        T.STAT  000032
LF.BWT= 000007        MO$29S= 000010        P.BLKS  000016        SF.ENA= 000100        T.ST2   000034
LF.ENA= 002000        M$HIGH= 000003        P.BUSY  000024        SF.LPB= 000004        T.ST3   000036
```

Table of contents

```
     5-   44     MACRO CALLS AND DATA
     6-  281     MANACP - NETWORK MANAGEMENT ACP
     6-  282     GETPKT - GET NEXT I/O PACKET (ACP RE-START AFTER FORK)
     7-  477     RETURN I/O ERRORS
     8-  517     MANINI - PROCESS NM.INI REQUESTS
     9-  597     MANOPR - PROCESS NM.OPR REQUESTS
    10-  629     MANCLN - PROCESS NM.CLN REQUESTS
    11-  656     FINREQ - FINISH REQUEST
    574 002502  001003                         BNE     35$                        ; BR IF NO
    575 002504  052764  000004  000002         BIS     #CF$VR2,C$FLAG(R4)         ; INDICATE V2.0 REQUEST
    576 002512  122763  002402  000012  35$:   CMPB    #NM.VR3,I.FCN(R3)          ; IS THIS REQUEST FROM V3.0 NCP?
    577 002520  001003                         BNE     40$                        ; BR IF NO
    578 002522  052764  000010  000002         BIS     #CF$VR3,C$FLAG(R4)         ; INDICATE V3.0 REQUEST
    579
    580 002530  116300  000025          40$:   MOVB    I.PRM+1(R3),R0             ; SET OPTION FOR CALL
    581 002534  042700  000017                 BIC     #MO$ENT,R0                 ; CLEAR OUT ENTITY BITS
    582 002540  012701  001410'                MOV     #IDBUFF,R1                 ; GET ID BUFFER
    583 002544  016302  000040                 MOV     I.PRM+14(R3),R2            ; ... LENGTH OF BUFFER
    584 002550  016363  000034  000024         MOV     I.PRM+10(R3),I.PRM(R3)     ; REARRANGE BUFFER POINTER FOR
    585 002556  016363  000036  000026         MOV     I.PRM+12(R3),I.PRM+2(R3);  ... COPYING DATA
    586 002564  016363  000042  000030         MOV     I.PRM+16(R3),I.PRM+4(R3);  ... TO USER BUFFER
    587 002572  012703  000734'                MOV     #BUFFER,R3                 ; SET BUFFER ADDRESS
    588 002576  012704  000244'                MOV     #CONTEX,R4                 ; ... CONTEXT ADDRESS
    589 002602                                 CALL    INIREQ                     ; SET UP TO INITIALIZE FOR THE REQUEST
    590 002606  103404                         BCS     45$                        ; IF CS, DON'T CALL THE FUNCTION
    591 002610                                 CALL    @R$INIT(R5)                ; CALL INITIALIZE ROUTINE
    592 002614  105700                         TSTB    R0                         ; SUCCESSFUL OPERATION ?
    593 002616  100002                         BPL     50$                        ; IF PL, YES
    594 002620  005267  175372          45$:   INC     DELFLG                     ; ELSE, MARK CONTEXT FOR DELETION
    595 002624  000444                  50$:   BR      FINREQ                     ; FINISH THE REQUEST
```

```
                                                                 D  5
MANACP      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

R$OPER     000006  G        #5-219
R$PC     = 000016  G        #5-232
R$R0     = 000002  G        #5-226
R$R1     = 000004  G        #5-227
R$R2     = 000006  G        #5-228
R$R3     = 000010  G        #5-229
R$R4     = 000012  G        #5-230
R$R5     = 000014  G        #5-231
R$SIZE     000010  G        #5-220      8-549
R$$MPL   = ******             5-171     6-373      6-382      6-457
R$$PRO   = ******             5-55      5-126      5-159      11-676
SDBAF      000156 RG        #5-149
SDBCT      000160 RG        #5-150
SDBNM      000162 RG        #5-151
SLTMA      000164 RG        #5-152
SLTNM      000166 RG        #5-153
SNAPT      000170 RG        #5-154
SRSTD      000040 RG        #5-100
STATUS     000230 RG        #5-268
STPCT      000042 RG        #5-101      6-365
TKTCB      000044 RG        #5-102      6-316      6-444
TSKLN2     000222 RG        #5-265     *6-387     *6-388      6-463      8-538      9-616      10-647
T.NAM    = ******   GX        6-330     6-332
T.RCVL   = ******   GX        6-348     6-445
T.ST3    = ******   GX        8-562
T.UCB    = ******   GX       *6-347     6-378     *6-378      8-564
T3.PRV   = ******   GX        8-562
UPDCTX   = ******   GX        8-568     9-619      10-650
U.CW2    = ******   GX        8-565
U2.PRV   = ******   GX        8-565
WBUFF    = ******   GX        6-434
W$ACP      000000  G        #5-186
W$CTX      000002  G        #5-187      9-618      10-649
W$OPR      000004  G        #5-188      9-623
W$SIZ      000006  G        #5-189
XAVL       000172 RG        #5-155
XBIAS      000174 RG        #5-156
ZTIME      000176 RG        #5-157
ZTIM2      000200 RG        #5-165
$ALOCB   = ******   GX        5-88
$BLXIO   = ******   GX        5-89
$CALLX   = ******   GX        5-169
$CCBAF   = ******   GX        5-103
$CCBAL   = ******   GX        5-104
$CCBCT   = ******   GX        5-105
$CCBGT   = ******   GX        5-106
$CCBNM   = ******   GX        5-108
$CCBRT   = ******   GX        5-107
$CEACC   = ******   GX        5-109
$COPT    = ******   GX        5-90
$CRUNM     000206 RG        #5-259
$CSBRT   = ******   GX        5-110
```

```
                                                                 E  5
MANACP      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 5
```

```
**FILE**ID**MANBUF
```

```
MM      MM     AAAAAA     NN      NN  BBBBBBBB    UU      UU  FFFFFFFFFF
MM      MM     AAAAAA     NN      NN  BBBBBBBB    UU      UU  FFFFFFFFFF
MMMM  MMMM   AA      AA   NN      NN  BB      BB  UU      UU  FF
MMMM  MMMM   AA      AA   NN      NN  BB      BB  UU      UU  FF
MM  MM  MM   AA      AA   NNNN    NN  BB      BB  UU      UU  FF
MM  MM  MM   AA      AA   NNNN    NN  BB      BB  UU      UU  FF
MM      MM   AA      AA   NN  NN  NN  BBBBBBBB    UU      UU  FFFFFFFF
MM      MM   AA      AA   NN  NN  NN  BBBBBBBB    UU      UU  FFFFFFFF
MM      MM   AAAAAAAAAA   NN    NNNN  BB      BB  UU      UU  FF
MM      MM   AAAAAAAAAA   NN    NNNN  BB      BB  UU      UU  FF
MM      MM   AA      AA   NN      NN  BB      BB  UU      UU  FF                ....
MM      MM   AA      AA   NN      NN  BB      BB  UU      UU  FF                ....
MM      MM   AA      AA   NN      NN  BBBBBBBB    UUUUUUUUUU  FF                ....
MM      MM   AA      AA   NN      NN  BBBBBBBB    UUUUUUUUUU  FF                ....


LL              SSSSSSSS    TTTTTTTTTT
LL              SSSSSSSS    TTTTTTTTTT
LL            SS                TT
LL            SS                TT
LL            SS                TT
LL            SS                TT
LL              SSSSSS          TT
LL              SSSSSS          TT
LL                    SS        TT
LL                    SS        TT
LL                    SS        TT
LL                    SS        TT
LLLLLLLLLL    SSSSSSSS          TT
LLLLLLLLLL    SSSSSSSS          TT
```

```
MANBUF - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:29
Table of contents
```

```
                                                                                 D  7
MANBUF - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:29  Page 11-2
Symbol table

G.TSVA= 000026            I.XIOP  000002         M$$FCS= 000000         PC.NRM= 000400         S$$YSZ= 007600
G.TSVL= 000030            I.XLEN= 000046         M$$MGE= 000000         PC.XAF= 000010         TKTCB = ****** GX
HEADR = ****** GX         I.XLNK  000000         M$$NET= 000000         PC.XIT= 000200         TSKLN2= ****** GX
IDBUFF= ****** GX         I.XMOD  000006         M$$OVR= 000000         PF.ALL= 177777         T$$KMG= 000000
IE.ABO= ****** GX         I.XPBF  000022         NMCLH = ****** GX      PF.INS= 000040         T$$MIN= 000000
INIBUF  000040RG      002 I.XPBL  000026         NMCL2 = ****** GX      PF.LOG= 000100         T.NAM = ****** GX
IOFIN = ****** GX         I.XPBV  000030         NUMUSR= ****** GX      PF.REQ= 000200         T.RCVL= ****** GX
IP.ABO= 000040            I.XRBF  000012         N$$ACC= 000001         POOLPT= ****** GX      UA.ACC= 000001
IP.FAK= 000020            I.XRBL  000016         N$$BUF= 000001         POOLSZ= ****** GX      UA.ALL= 000400
IP.PND= 000100            I.XTCB  000004         N$$LDV= 000001         P$$P45= 000000         UA.CAL= 000100
IP.UMR= 000200            I.XTMO  000020         N$$MCP= 000001         P$$WRD= 000000         UA.COM= 000200
I$$RAR= 000000            I.XTTB  000032         N$$MLL= 000001         Q$$OPT= 000010         UA.ECH= 000004
I$$RDN= 000000            I.XTTL  000036         N$$MOV= 000010         RBUFF   000434RG       UA.PRO= 000002
I.AST   000022            K$$CNT= 177546         N$$NCT= 000001         READ    000444RG       UA.PUT= 000040
I.ATIL= 000044            K$$CSR= 177546         N$$PEM= 000001         RELOC = ****** GX      UA.SPE= 000020
I.EFN   000003            K$$LDC= 000000         O.AST   000006         REXAST= ****** GX      UA.TRA= 002000
I.FCN   000012            K$$TPS= 000074         O.EFN   000010         RID     000452RG       UA.TRN= 001000
I.IOSB  000014            LDBRT = ****** GX      O.ESB   000012         RX$IAS= 000003         UA.TYP= 000010
I.LGTH= 000044            LD$LP = 000000         O.LGTH= 000034         RX$11D= 000000         UPDCTX  000346RG
I.LNK   000000            LSTCTX= ****** GX      O.LNK   000000         RX$11M= 000001         V$$CTR= 001000
I.LN2   000006            L$$ASG= 000000         O.MCRL  000002         RX$11S= 000002         WBUFF   000526RG
I.PRI   000002            L$$DRV= 000000         O.PTCB  000004         R$OPER= ****** GX      W$SIZ = ****** GX
I.PRM   000024            L$$P11= 000001         O.STAT  000014         R$$DER= 000000         X$$DBT= 000000
I.TCB   000004            L$$11R= 000000         PC.ALF= 000004         R$$K11= 000001         $SAVAL= ****** GX
I.UCB   000010            MN$VOL= 000000         PC.ALM= 001000         R$$SND= 000000         $$$OST= 000004
I.XDBF  000040            M$$CRB= 000124         PC.HIH= 000001         R$$11M= 000000         .$$$$.= 000034
I.XDBL  000044            M$$CRX= 000000         PC.LOW= 000002         S$$WRG= 000000

. ABS.  177776    000    (RW,I,GBL,ABS,OVR)
        000572    001    (RW,I,LCL,REL,CON)
$$$XXX  000316    002    (RW,I,LCL,REL,OVR)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 46
Work  file writes: 60
Size of work file: 20423 Words  ( 80 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:22.40
SY:MANBUF.V2,[135,134]MANBUF/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANBUF
```

```
                                                                E  7
MANBUF      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 1
```

```
                                        D 8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7
DECBYE - LOGOUT CLEANUP FOR DECNET

    119                                                     .SBTTL  DECBYE - LOGOUT CLEANUP FOR DECNET
    120                                             ;+
    121                                             ; **-DECBYE-LOGOUT CLEANUP FOR DECNET
    122                                             ;
    123                                             ; THIS ROUTINE DEALLOCATES THE ALIASES FOR A USER ON LOGOUT.
    124                                             ;
    125                                             ; INPUTS:
    126                                             ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
    127                                             ;       R3 = ADDRESS OF INPUT/OUTPUT BUFFER
    128                                             ;       R4 = ADDRESS OF CONTEXT AREA
    129                                             ;
    130                                             ; OUTPUTS:
    131                                             ;       ALL ALIASES FOR THIS USER ARE CLEARED
    132                                             ;
    133                                             ; REGISTERS:
    134                                             ;       R3, R4, R5 ARE PRESERVED
    135                                             ;-
    136
    137 000040                                      DECBYE:
    138 000040  012764  000001  000000                      MOV     #AF.INI,A$FLG(R4)       ; INDICATE WE'VE BEEN INITIALIZED
    139 000046  112764  000100  000002                      MOVB    #MO$CLE,A$OPT(R4)       ; SET OPERATION TO ''CLEAR''
    140 000054  112764  177777  000011                      MOVB    #MF$KNO,A$TYP(R4)       ; SET TYPE TO ''KNOWN''
    141 000062  017700  000000G                             MOV     @TKTCB,R0               ; GET ADDRESS OF OUR TCB        ; RJK01
    142 000066  016064  000000G 000020                      MOV     T.UCB(R0),A$SCO(R4)     ; SET SCOPE TO ISSUING TASK'S TI:
    143 000074                                              CALLR   ALCHOP                  ; CLEAR THE ALIASES
    144
    145         000001                                      .END
```

```
                                        E 8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-1
Symbol table
```

```
                                                                D 9
MANFRK - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:30  Page 7-2
Symbol table

AB.NPV= 000001                   A.SLEN  000024                   G.CNT   000004                   N$$BUF= 000001                   TS.NRP= 010000
AB.TYP= 000002                   A.SMAP  000020                   G.EFLG  000006                   N$$LDV= 000001                   TS.OUT= 000400
AF.AST= 000010                   A.STA   000015                   G.GRP   000002                   N$$MCP= 000001                   TS.RDN= 040000
AF.ESQ= 000020                   A.TRAN  000022                   G.LGTH= 000012                   N$$MLL= 000001                   TS.RUN= 004000
AF.LCK= 000040                   BUFFER= ****** GX                G.LNK   000000                   N$$MOV= 000010                   TS.STP= 001000
AF.MDE= 000200                   CF$FDN= ****** GX                G.STAT  000003                   N$$NCT= 000001                   T$$KMG= 000000
AF.NOT= 000002                   CM.CDS= 000004                   IP.ABO= 000040                   N$$PEM= 000001                   T$$MIN= 000000
AF.OOB= 000004                   CM.CEN= 000003                   IP.FAK= 000020                   O.AST   000006                   T.ACTL  000052
AF.QUE= 000100                   CM.ELM= 000005                   IP.PND= 000100                   O.EFN   000010                   T.ASTL  000016
AF.XCC= 000001                   CM.EXT= 000006                   IP.UMR= 000200                   O.ESB   000012                   T.ATT   000062
AK.BUF= 000200                   CM.IND= 000002                   ISTACK= ****** GX                O.LGTH= 000034                   T.CPCB  000004
AK.GBI= 000202                   CM.INE= 000001                   I$$RAR= 000000                   O.LNK   000000                   T.DPRI  000040
AK.GGF= 000303                   CM.LKT= 000007                   I$$RDN= 000000                   O.MCRL  000002                   T.EFLG  000022
AK.OCB= 000201                   CM.MSG= 000011                   I.AST   000022                   O.PTCB  000004                   T.IOC   000003
AS.CAA= 000006                   CM.RMT= 000010                   I.ATTL= 000044                   O.STAT  000014                   T.LBN   000041
AS.DIS= 000002                   CONTEX= ****** GX                I.EFN   000003                   PC.ALF= 000004                   T.LDV   000044
AS.DLT= 000001                   CP.DSB= 000010                   I.FCN   000012                   PC.ALM= 001000                   T.LNK   000000
AS.FPA= 000001                   CP.EXT= 000400                   I.IOSB  000014                   PC.HIH= 000001                   T.MXSZ  000050
AS.PFA= 000004                   CP.LGO= 000004                   I.LGTH= 000044                   PC.LOW= 000002                   T.NAM   000006
AS.RCA= 000002                   CP.MSG= 000002                   I.LNK   000000                   PC.NRM= 000400                   T.OFF   000066
AS.REA= 000005                   CP.NIO= 000100                   I.LN2   000006                   PC.XAF= 000010                   T.PCB   000046
AS.RRA= 000003                   CP.NUL= 000001                   I.PRI   000002                   PC.XIT= 000200                   T.PRI   000002
A$$CHK= 000000                   CP.PRV= 000020                   I.PRM   000024                   PF.ALL= 177777                   T.RCVL  000012
A$$CPS= 000000                   CP.RST= 000200                   I.TCB   000004                   PF.INS= 000040                   T.RRFL  000072
A$$PRI= 000000                   CP.SGL= 000040                   I.UCB   000010                   PF.LOG= 000100                   T.SAST  000054
A$$TRP= 000000                   CTXHD = ****** GX                I.XDBF  000040                   PF.REQ= 000200                   T.SRCT  000071
A.ACC   000014                   CTXSIZ= ****** GX                I.XDBL  000044                   P$$P45= 000000                   T.STAT  000032
A.ACCE  000000                   CURCTX= ****** GX                I.XIOP  000002                   P$$WRD= 000000                   T.ST2   000034
A.AST   000006                   CURIOP= ****** GX                I.XLEN= 000046                   Q$$OPT= 000010                   T.ST3   000036
A.BYT   000004                   C$FLAG= ****** GX                I.XLNK  000000                   READ  = ****** GX                T.TCBL  000030
A.CALL  000024                   C$SP  = ****** GX                I.XMOD  000006                   R$$DER= 000000                   T.TIO   000057
A.CBL   000002                   C$STAT= ****** GX                I.XPBF  000022                   R$$K11= 000001                   T.TKSZ  000060
A.CONN  000012                   C$$CKP= 000000                   I.XPBL  000026                   R$$SND= 000000                   T.UCB   000026
A.DEQU  000002                   C$$ORE= 000400                   I.XPBV  000030                   R$$11M= 000000                   T2.ABO= 000100
A.DIS   000002                   C$$RSH= 177564                   I.XRBF  000012                   SS$LOA= 020000                   T2.AST= 100000
A.DISC  000014                   C.PCPL  000011                   I.XRBL  000016                   SS$NDA= 040000                   T2.CAF= 000400
A.DQSR  177776                   C.PDPL  000010                   I.XTCB  000004                   SS$OUT= 010000                   T2.CHK= 020000
A.FLEN  000011                   C.PNAM  000002                   I.XTMO  000020                   STATUS= ****** GX                T2.CKD= 010000
A.IBUF  000014                   C.PRMT  000012                   I.XTTB  000032                   S$DISK  000000                   T2.DST= 040000
A.ILEN  000016                   C.PSTS  000006                   I.XTTL  000036                   S$DOWN  000010                   T2.FXD= 002000
A.IMAP  000012                   C.PTCB  000000                   K$$CNT= 177546                   S$LOAD  000002                   T2.HLT= 000200
A.INPU  000006                   D$$BUG= 177514                   K$$CSR= 177546                   S$NAME  000014                   T2.REX= 001000
A.IOS   000026                   D$$ISK= 000000                   K$$LDC= 000000                   S$NEXT  000012                   T2.SEF= 004000
A.KSR5  177774                   D$$L11= 000001                   K$$TPS= 000074                   S$SIZE  000004                   T2.SPN= 000004
A.LIN   000012                   D$$YNC= 000000                   LD$LP = 000000                   S$STAT  000000                   T2.STP= 000020
A.MAS   000004                   D$$YNM= 000000                   LSTCTX= ****** GX                S$UP    000006                   T2.WFR= 000001
A.NPR   000010                   E$$XPR= 000000                   L$$ASG= 000000                   S$WIND  000020                   T3.ACP= 100000
A.NUM   000010                   FDNCNT= ****** GX                L$$DRV= 000000                   S$$WRG= 000000                   T3.CAL= 000100
A.OUTP  000010                   FORK    000000RG                 L$$P11= 000001                   S$$TSZ= 007600                   T3.CLI= 001000
A.PLEN= 000016                   FRKDON  000202RG                 L$$11R= 000000                   TKTCB = ****** GX                T3.GFL= 000010
A.POWE  000004                   F$$LVL= 000001                   M$$CRB= 000124                   TSKLN2= ****** GX                T3.MCR= 004000
A.PRM   000012                   GETPKT= ****** GX                M$$CRX= 000000                   TS.BLK= 171700                   T3.NET= 000020
A.PROC  000020                   GS.DEL= 000001                   M$$FCS= 000000                   TS.CKP= 000200                   T3.NSD= 000200
A.RECE  000016                   G$$TPP= 000000                   M$$MGE= 000000                   TS.CKR= 000100                   T3.PMD= 040000
A.REL   000000                   G$$TSS= 000000                   M$$NET= 000000                   TS.EXE= 100000                   T3.PRV= 010000
A.RES = 000032                   G$$TTK= 000000                   M$$OVR= 000000                   TS.HLD= 002000                   T3.REM= 020000
A.SBUF  000022                   G$$WRD= 000000                   N$$ACC= 000001                   TS.MSG= 020000                   T3.ROV= 000040
```

```
                                                                E 9
MANFRK - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:30  Page 7-3
Symbol table
```

```
    311 000512  000402                    BR      PRID
    312 000514  012704            PRREIN::MOV     (PC)+,R4
    313 000516     005                    .BYTE   ME$PRO
    314 000517     224                    .BYTE   MF$REA!CM$RSX
    315 000520                    ALID:
    316 000520                    CIID:
    317 000520                    LIID:
    318 000520  111146            PRID:   MOVB    (R1),-(SP)              ; GET LENGTH OF STRING
    319 000522  100001                    BPL     10$                     ; IF PL, COUNTED STRING
    320 000524  005016                    CLR     (SP)                    ; ELSE, ONLY 1 BYTE
    321 000526  005216            10$:    INC     (SP)                    ; INCLUDE COUNT IN COUNTED STRING
    322 000530  000467                    BR      100$                    ; ON TO COMMON CODE
    323                           ;
    324                           ; LOGGING INITIALIZE
    325                           ;
    326 000532  012704            LOCHIN::MOV     (PC)+,R4
    327 000534     002                    .BYTE   ME$LOG
    328 000535     023                    .BYTE   MF$CHA
    329 000536  000402                    BR      LOID
    330 000540  012704            LOREIN::MOV     (PC)+,R4
    331 000542     002                    .BYTE   ME$LOG
    332 000543     024                    .BYTE   MF$REA
    333 000544  012746  000001    LOID:   MOV     #1,-(SP)                ; LOGGING ID LENGTH IS ALWAYS 1 BYTE
    334 000550  000457                    BR      100$
    335                           ;
    336                           ; MODULE INITIALIZE
    337                           ;
    338 000552  012704            MOCHIN::MOV     (PC)+,R4
    339 000554     004                    .BYTE   ME$MOD
    340 000555     023                    .BYTE   MF$CHA
    341 000556  000405                    BR      MOID
    342 000560  012704            MOREIN::MOV     (PC)+,R4
    343 000562     004                    .BYTE   ME$MOD
    344 000563     024                    .BYTE   MF$REA
    345 000564  000402                    BR      MOID
    346 000566  012704            MOZEIN::MOV     (PC)+,R4
    347 000570     004                    .BYTE   ME$MOD
    348 000571     025                    .BYTE   MF$ZER
    349 000572  111146            MOID:   MOVB    (R1),-(SP)              ; ALWAYS COUNTED STRING
    350 000574  005216                    INC     (SP)                    ; INCLUDE FORMAT BYTE IN LENGTH
    351 000576  000444                    BR      100$
    352                           ;
    353                           ; NODE INITIALIZE
    354                           ;
    355 000600  012704            NOCHIN::MOV     (PC)+,R4
    356 000602     000                    .BYTE   ME$NOD
    357 000603     023                    .BYTE   MF$CHA
    358 000604  000405                    BR      NOID
    359 000606  012704            NOREIN::MOV     (PC)+,R4
    360 000610     000                    .BYTE   ME$NOD
    361 000611     024                    .BYTE   MF$REA
    362 000612  000402                    BR      NOID
    363 000614  012704            NOZEIN::MOV     (PC)+,R4
    364 000616     000                    .BYTE   ME$NOD
    365 000617     025                    .BYTE   MF$ZER
    366 000620  111146            NOID:   MOVB    (R1),-(SP)              ; ASSUME COUNTED STRING
    367 000622  001002                    BNE     20$                     ; IF NE, NOT AN ADDRESS
```

```
          .TITLE  MANTRA - TRANSLATE NAMES AND ADDRESS FOR MANACP
          .IDENT  /V05.00/
;
; COPYRIGHT (C) 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       NETWORK MANAGEMENT ACP NAME AND ADDRESS TRANSLATION QIO ROUTINES
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

Table of contents

```
     5-   49    MACRO CALLS AND IMPURE DATA
     6-   73    INIACP - INITIALIZE ACP
     7-   99    INIREQ - INITIALIZE ACP FOR A REQUEST
```

```
                                        .SBTTL  MACRO CALLS AND LOCAL DEFINITIONS
                                ;
                                ; MACRO LIBRARY CALLS
                                ;
                                        .MCALL  MANDF$

 000000                                 MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS

          000200                        CM$RSX=200                      ; DEFINE RSX SPECIFIC FUNCTION FLAG
                                ;
                                ; LOCAL MACRO DEFINTIONS
                                ;
                                        .MACRO  MNFUN$  FUNC=0,OPT=0,INIT=0,OPER=0,FLAGS=0
                                        .BYTE   FUNC
                                        .BYTE   OPT
                                        .BYTE   FLAGS
                                        .BLKB   1
                                        .IIF NB INIT .WORD INIT
                                        .IIF NB OPER .WORD OPER
                                        .ENDM   MNFUN$
                                        .ENABL  LC
```

```
                                      .SBTTL  Macro calls and definitions

                                      .MCALL  CALLX,CEACC$,PHBDF$,RDTDF$,RESMAP,SAVMAP,TCBDF$

 000000                               PHBDF$                           ; Define PSI home block offsets
 000000                               RDTDF$                           ; Define remote DTE descriptors
 000000                               TCBDF$                           ; Define TCB offsets

 000000                               .PSECT  $$$BYE
 000000  000000'                      .WORD   PSIBYE                   ; CLEANUP FOR PSI
 000000                               .PSECT                           ; RETURN TO CODE PSECT

```

```
000374  103431                      BCS     30$                     ; IF CS, ERROR
000376  010103                      MOV     R1,R3                   ; POINT PAST NAME FIELD
000400  010400              20$:    MOV     R4,R0                   ; COPY CONTEXT BLOCK POINTER
000402  062700  000030              ADD     #A$DES+6,R0             ; POINT TO DEST. ACCESS CONTROL BUFFER

                            ;
                            ; CHECK AND SAVE ACCESS CONTROL FIELDS
                            ;
000406  012705  000020              MOV     #MAXID,R5               ; SET MAXIMUM LENGTH FOR USER ID
000412                              CALL    CHKACC                  ; LENGTH CHECK IT AND MOVE IT
000416  103420                      BCS     30$                     ; IF CS, ERROR
000420  012705  000010              MOV     #MAXPAS,R5              ; SET MAXIMUM LENGTH FOR PASSWORD
000424                              CALL    CHKACC                  ; LENGTH CHECK IT AND MOVE IT
000430  103413                      BCS     30$                     ; IF CS, ERROR
000432  012705  000020              MOV     #MAXACC,R5              ; SET MAXIMUM LENGTH FOR ACCOUNT
000436                              CALL    CHKACC                  ; LENGTH CHECK IT AND MOVE IT
000442  103406                      BCS     30$                     ; IF CS, ERROR
                            ;
                            ; AT THIS POINT, A DUMMY ALIAS BLOCK HAS BEEN CREATED AND SAVED AT
                            ; A$ALI-2(R4). THE LENGTH OF THE ALIAS BLOCK IS CALCULATED, AND
                            ; SAVED FOR LATER ALLOCATION AND COPYING.
                            ;
000444  160400                      SUB     R4,R0                   ; CALC HOW FAR WE ARE INTO CONTEXT AREA
000446  162700  000010              SUB     #<A$ALI-2>,R0           ; CALC LENGTH OF ALIAS BLOCK
000452  110064  000010              MOVB    R0,A$NUM(R4)            ; SAVE THE LENGTH
000456  000403                      BR      40$                     ; AND RETURN

000460  012700  177760      30$:    MOV     #ME.PVA,R0              ; SET PARAMETER VALUE ERROR
000464  000261                      SEC
000466                      40$:    RETURN
```

```
                                ;

                                        .IF DF  R$$PRO                  ;; PRO/DECNET

                                30$:    MOV     @DECPT,R1               ;; POINT TO DECNET HOME BLOCK
                                        TST     D$ALNA(R1)              ;; CHECK # OF AVAILABLE ALIAS ENTRIES
                                        BGT     33$                     ;; IF GT, SOME AVAILABLE IN NT.CFG
                                        MOV     #ME.RES,2+BIAS(SP)      ;; ELSE, MAX ALIAS ENTRIES ERROR
                                        BR      90$                     ;; DECLARED AND RETURN
                                33$:    DEC     D$ALNA(R1)              ;; DECREMENT # OF AVAILABLE ALIASES
                                        MOVB    A$NUM(R4),R1            ;; GET LENGTH OF BLOCK TO ALLOCATE

                                        .IFF

000152  116401  000010          30$:    MOVB    A$NUM(R4),R1            ;; GET LENGTH OF BLOCK TO ALLOCATE

                                        .ENDC

000156  010105                          MOV     R1,R5                   ;; SAVE A COPY OF THE LENGTH
000160                                  CALL    ALOCB                   ;; TRY TO ALLOCATE ALIAS BLOCK
000164  103004                          BCC     40$                     ;; IF CC, GOT IT - CONTINUE
000166  012766  177761  000004          MOV     #ME.RES,2+BIAS(SP)      ;; ELSE, DECLARE RESOURCE ERROR

                                        .IF DF  R$$PRO                  ;; PRO/DECNET

                                ;
                                ; IF ALLOCATION ERROR SET # OF AVAILABLE ALIASES TO PREVIOUS VALUE
                                ;
                                        MOV     @DECPT,R1               ;; POINT TO DECNET HOME BLOCK
                                        INC     D$ALNA(R1)              ;; INCREMENT # OF AVAILABLE ALIASES

                                        .ENDC   ; DF R$$PRO

000174  000455                          BR      90$                     ;; AND RETURN
                                ;
                                ; COPY DATA INTO NEW ALIAS BLOCK
                                ;
000176  010046                  40$:    MOV     R0,-(SP)                ;; R0 IS UNMAPPED ADDRESS/ MAP TO ADDRESS
000200                                  CALL    @CEACC                  ;; MAP TO BLOCK
000204  012601                          MOV     (SP)+,R1                ;; MAPPED ALIAS BLOCK ADDRESS
000206  010402                          MOV     R4,R2                   ;; POINT TO THE CONTEXT AREA
000210  062702  000010                  ADD     #A$ALI-2,R2             ;; POINT TO SAVED ALIAS BLOCK DATA-2
000214  112221                  50$:    MOVB    (R2)+,(R1)+             ;; COPY ALIAS DATA INTO BLOCK
000216  005305                          DEC     R5                      ;; LOOP FOR COUNT
000220  001375                          BNE     50$                     ;; ...
                                ;
                                ; LINK NEW BLOCK INTO LIST, SORTED BY SCOPE
                                ;
000222  010046                          MOV     R0,-(SP)                ;; SAVE THE NEW ANB ADDRESS
000224  017701  000000G                 MOV     @DECPT,R1               ;; POINT TO DEC BLOCK

                                        .IF DF R$$PRO                   ;; PRO DECNET

                                        BIS     #DF.AFN,D$STAT(R1)      ;; SET UPDATE ALIAS FILE BIT

                                        .ENDC
                                        .SBTTL  LOGEVT  - Log a Network Event
                                ;+
                                                        *** LOGEVT ***
                                ;
                                ; Function:     To log microcode loading events to the network using the
                                ;               NM: network management pseudo-device.
                                ;
                                ; Call Format:  CALL    LOGEVT
                                ;
                                ; Inputs:       R2 = address of specific error ASCII string
                                ;               PDVIDX = PDV index for device driver
                                ;               CTLNUM = device controller number
                                ;
                                ; Outputs:      DECnet event is logged to the event sink for the node
                                ;               this program is running on.
                                ;
                                ; Registers:    R0, R1, R2 and R5 destroyed
                                ;
                                ;-

000624  010205                  LOGEVT: MOV     R2,R5                   ; Copy address of input string
000626  016746  000162'                 MOV     CTLNUM,-(SP)            ; Put controller number on stack
000632  017701  000614'                 MOV     @PDVTA,R1               ; Get address of PDV table
000636  066701  000160'                 ADD     PDVIDX,R1               ; Advance to PDV address of driver
000642  011101                          MOV     (R1),R1                 ; Get driver's PDV address
000644  016146  000004                  MOV     Z.NAM(R1),-(SP)         ; Put RAD50 process name on stack

000650  012700  000326'                 MOV     #EVTBLK+E$DATA,R0       ; Get address of event output data
000654  012701  000000'                 MOV     #MLDERR,R1              ; Get $EDMSG edit instructions address
000660  010602                          MOV     SP,R2                   ; Get $EDMSG argument block address
000662                                  CALL    $EDMSG                  ; Form first ASCII string
000666  105020                          CLRB    (R0)+                   ; Make it an ASCIZ string
000670  022626                          CMP     (SP)+,(SP)+             ; Clean off the stack

000672  112520                  10$:    MOVB    (R5)+,(R0)+             ; Append a byte from input string
000674  001376                          BNE     10$                     ; End of string is marked by zero
000676  162700  000306'                 SUB     #EVTBLK,R0              ; Get event buffer size, log, and exit

000702                                  QIOW$S  #IO.WLB,#EVTLUN,#EVTEFL,,,,<#EVTBLK,R0,#ETYPCL,#0>
000752                                  RETURN
```

```
                                                                  E  2
MLD     - Microcode Loader       MACRO V05.03b Wednesday 28-Aug-85 11:31  Page 18-4
Symbol table

T.TCBL  000030          T3.ACP= 100000           X$$DBT= 000000          ZF.X3P= 000000          $EDMSG= ****** GX
T.TIO   000057          T3.CAL= 000100           ZF.COU= 001000          ZS.ASN= 100000          $LDBGT= ****** GX
T.TKSZ  000060          T3.CLI= 001000           ZF.DDM= 000001          ZS.BSY= 140000          $LDBRT= ****** GX
T.UCB   000026          T3.GFL= 000010           ZF.DIA= 004000          Z.AVL   000014          $NTUIC= ****** GX
T2.ABO= 000100          T3.MCR= 004000           ZF.DLC= 000002          Z.DAT   000016          $PARHD= ****** GX
T2.AST= 100000          T3.NET= 000020           ZF.DVP= 100000          Z.DSP   000000          $PDSPL= ****** GX
T2.CAF= 000400          T3.NSD= 000200           ZF.INI= 040000          Z.FLG   000010          $PDVTA= ****** GX
T2.CHK= 020000          T3.PMD= 040000           ZF.KMX= 000020          Z.LEN = 000016          $QRMVF= ****** GX
T2.CKD= 010000          T3.PRV= 010000           ZF.LLC= 000004          Z.LLN   000006          $RDBSZ= ****** GX
T2.DST= 040000          T3.REM= 020000           ZF.LMC= 000100          Z.MAP   000020          $SLTMA= ****** GX
T2.FXD= 002000          T3.ROV= 000040           ZF.MAN= 020000          Z.NAM   000004          $STDDM= ****** GX
T2.HLT= 000200          T3.RST= 000400           ZF.MFL= 000010          Z.PCB   000012          $TKTCB= ****** GX
T2.REX= 001000          T3.SLV= 002000           ZF.MTM= 000400          Z.SCH   000007          $$$    = 000062
T2.SEF= 004000          T3.SWS= 000002           ZF.MUX= 000040          $BLXIO= ****** GX       $$$ARG= 000002
T2.SPN= 000004          V$$CTR= 001000           ZF.PSE= 002000          $CAT5 = ****** GX       $$$OST= 000004
T2.STP= 000020          WAIT    000140R      002 ZF.SLI= 010000          $CCBRT= ****** GX       .$$$$.= 000034
T2.WFR= 000001          WORKSP  000166R      002 ZF.TIM= 000200

. ABS.  177776     000     (RW,I,GBL,ABS,OVR)
        000000     001     (RW,I,LCL,REL,CON)
MLDDAT  000624     002     (RW,D,LCL,REL,OVR)
MLDCOD  001456     003     (RO,I,LCL,REL,OVR)
Errors detected:   0

*** Assembler statistics


Work  file  reads: 125
Work  file writes: 96
Size of work file: 23690 Words  ( 93 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:26.41
DB2:MLD11S.T34,[132,134]MLD11S/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[132,10]T34,MLD
```

```
                                                        F  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF  04.00
```

```
        .TITLE  MANACP - NETWORK MANAGEMENT ACP
        .IDENT  /V05.00/
;
; COPYRIGHT (C) 1981, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       NETWORK MANAGEMENT ACP FUNCTION DISPATCHING
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;

```

```
                                                .SBTTL  MANOPR - PROCESS NM.OPR REQUESTS
                                        ;+
                                        ; **-MANOPR-PROCESS NM.OPR REQUESTS
                                        ;
                                        ; THIS ROUTINE CALLS THE CORRECT OPERATE ROUTINE FOR THE REQUESTED
                                        ; FUNCTION.
                                        ;
                                        ; INPUTS:
                                        ;       R3 = ADDRESS OF I/O PACKET
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = I/O STATUS TO RETURN
                                        ;       R1 = NUMBER OF BYTES TO TRANSFER TO USER BUFFER
                                        ;
                                        ; REGISTERS:
                                        ;       R3 IS PRESERVED
                                        ;-
 002626                                 MANOPR:
 002626  016705  175370                         MOV     TSKLN2,R5               ; GET WINDOW BLOCK ADDRESS
 002632  001611                                 BEQ     NLNERR                  ; IF EQ, NO FILE ON LUN
 002634  016567  000002  175350                 MOV     W$CTX(R5),CURCTX        ; SET CONTEXT AREA ADDRESS
 002642                                         CALL    UPDCTX                  ; RESTORE THE CONTEXT AREA
 002646  016302  000030                         MOV     I.PRM+4(R3),R2          ; GET LENGTH OF BUFFER
 002652  012703  000734'                        MOV     #BUFFER,R3              ; SET BUFFER ADDRESS
 002656  012704  000244'                        MOV     #CONTEX,R4              ; ... CONTEXT ADDRESS
 002662                                         CALL    @W$OPR(R5)              ; CALL OPERATE ROUTINE
 002666  120027  177600                         CMPB    R0,#ME.DON              ; LAST TIME THROUGH OPERATE ?
 002672  001002                                 BNE     10$                     ; IF NE, NO
 002674  005267  175316                         INC     DELFLG                  ; ELSE, MARK CONTEXT FOR DELETION
 002700  000416                         10$:    BR      FINREQ                  ; FINISH THE REQUEST
```

```
                                                                      E  5
MANACP      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 5
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE              REFERENCES

$DDMAN  = ******  GX        5-111
$DEACB  = ******  GX        5-91
$DECPT  = ******  GX        5-112
$DREXT  = ******  GX        5-92
$EXRQN  = ******  GX        5-93
$EXRQP  = ******  GX        5-94
$FILHD  = ******  GX        5-113
$FMASK  = ******  GX        5-95
$HEADR  = ******  GX        5-96
$IOFIN  = ******  GX        5-97
$LDBAF  = ******  GX        5-114
$LDBGT  = ******  GX        5-115
$LDBRT  = ******  GX        5-116
$LGCON  = ******  GX        5-117
$LGFNB  = ******  GX        5-118
$LGMON  = ******  GX        5-119
$LGSTT  = ******  GX        5-120
$LGUIC  = ******  GX        5-121
$LLCRS  = ******  GX        5-122
$LLCTA  = ******  GX        5-123
$MOULS  = ******  GX        5-98
$NMCLH  = ******  GX        5-124
$NMCL2  = ******  GX        5-132
$NMCRS  = ******  GX        5-136
$NMLST  = ******  GX        5-137
$NTLPT  = ******  GX        5-138
$OBJHD  = ******  GX        5-139
$PDVID  = ******  GX        5-140
$PDVNM  = ******  GX        5-141
$PDVTA  = ******  GX        5-142
$PRUNM    000204 RG        #5-258
$PSIPT  = ******  GX        5-143
$RDBAF  = ******  GX        5-144
$RDBCT  = ******  GX        5-145
$RDBNM  = ******  GX        5-146
$RDBSZ  = ******  GX        5-147
$RDBTH  = ******  GX        5-148
$RELOC  = ******  GX        5-99
$SDBAF  = ******  GX        5-149
$SDBCT  = ******  GX        5-150
$SDBNM  = ******  GX        5-151
$SLTMA  = ******  GX        5-152
$SLTNM  = ******  GX        5-153
$SNAPT  = ******  GX        5-154
$SRSTD  = ******  GX        5-100
$STPCT  = ******  GX        5-101
$TKTCB  = ******  GX        5-102
$XAVL   = ******  GX        5-155
$XBIAS  = ******  GX        5-156
$ZTIME  = ******  GX        5-157
$ZTIM2  = ******  GX        5-165
```

```
                                                                      F  5
MANACP      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 6
```

E 6

Table of contents

```
     5-   44     MACRO CALLS AND LOCAL DATA
     6-   59     CONTEXT AREA POOL AND INITIALIZATION CODE
     7-  153     ALLCTX - ALLOCATE A CONTEXT AREA
     8-  244     DEACTX - DEALLOCATE CONTEXT AREA
     9-  284     UPDCTX - UPDATE THE CURRENT CONTEXT
    10-  325     RBUFF - READ THE USER'S BUFFER INTO "BUFFER"
    10-  326     READ - READ A SPECIFIED BUFFER INTO "BUFFER"
    10-  327     RID - READ THE USER'S ID STRING INTO "IDBUFF"
    11-  375     WBUFF - WRITE "BUFFER" INTO USER'S BUFFER
```

```
MA

SY

SY

AA
AL
BL
BU
BU
CC
CO
CS
CT
CT
CT
CU
CU
C$
C$
C$
C.
C.
DD
DE
FE
FM
G.
G.
HE
ID
IE
IN
IO
I.
I.
LD
LS
MN
NM
NM
NU
PO
PO
RB
RE
RE
RE
RE
RI
R$
TK
TS
T.
T.
UP
WE
W$
```

F 6

```
                                                                        E  7
MANBUF      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE              REFERENCES

AALOCB  = ******  GX        7-232
ALLCTX    000000 RG        #7-184
BLXIO   = ******  GX        10-371       11-404
BUFFER  = ******  GX        10-355       11-397
BUFTBL    000306 R          6-119       #6-144
CCBRT   = ******  GX        6-144
CONTEX  = ******  GX        7-226        9-307        9-318
CSBRT   = ******  GX        6-147
CTXHD   = ******  GX        7-185
CTXPOO    000000 R         #6-80         6-126        6-127        6-131        6-132
CTXSIZ  = ******  GX        7-227        9-320
CURCTX  = ******  GX       *7-224        8-265       *8-272        9-310        9-315
CURIOP  = ******  GX        8-274
C$DATA  = ******  GX        9-312        9-317
C$FLAG  = ******  GX        7-188       *7-221       *7-223       *8-271
C$SIZE  = ******  GX        7-193        7-195        7-216        7-218
C.BID     000003            6-117
C.BUF2    000024            6-110        6-112
DDEACB  = ******  GX        8-279
DEACTX    000264 RG        #8-263
FE.CEX  = ******  GX        6-104
FMASK   = ******  GX        6-104
G.TSTN  = 000000            6-127
G.TSTS  = 000032            6-131
HEADR   = ******  GX        7-234
IDBUFF  = ******  GX        10-360
IE.ABO  = ******  GX        6-98
INIBUF    000040 RG        #6-83
IOFIN   = ******  GX        6-100
I.PRM     000024            10-354       10-359       11-402       11-403
I.TCB     000004            6-90        *6-94
LDBRT   = ******  GX        6-145
LSTCTX  = ******  GX        9-308
MN$VOL  = 000000           #5-56         6-103        6-142
NMCLH   = ******  GX        6-106
NMCL2   = ******  GX        6-116
NUMUSR  = ******  GX       *7-225       *8-268
POOLPT  = ******  GX       *6-126        7-196        7-217       *7-218
POOLSZ  = ******  GX       *6-133        7-193        7-202       *7-214       *7-216
RBUFF     000434 RG        #10-353
READ      000444 RG        #10-355
RELOC   = ******  GX        10-362       11-398
REXAST  = ******  GX        6-134
RID       000452 RG        #10-358
R$OPER  = ******  GX        7-241
TKTCB   = ******  GX        6-85         7-239
TSKLN2  = ******  GX       *7-238        8-275       *8-277
T.NAM   = ******  GX        6-110        6-112
T.RCVL  = ******  GX        6-86
UPDCTX    000346 RG        #9-305
WBUFF     000526 RG        #11-394
W$SIZ   = ******  GX        7-231        8-278
```

```
                                                                        F  7
MANBUF      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 2
```

```
                                                          E  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-1
Symbol table

AB.NPV= 000001          A.LIN   000012              G$$WRD= 000000          MB$FUL  000002          MC$NRE  001200
AB.TYP= 000002          A.MAS   000004              G.CNT   000004          MB$MIX  000002          MC$NRT  001166
AF.AST= 000010          A.NPR   000010              G.EFLG  000006          MB$ONE  000001          MC$NVR  001642
AF.ESQ= 000020          A.NUM   000010              G.GRP   000002          MB$RXO  000001          MC$P08  000000
AF.INI= 000001          A.OUTP  000010              G.LGTH= 000012          MB$TXO  000000          MC$P11  000001
AF.LCK= 000040          A.PLEN= 000016              G.LNK   000000          MB$ZER  000000          MC$RFL  002046
AF.MDE= 000200          A.POWE  000004              G.STAT  000003          MC$BAB  006073          MC$SBU  002051
AF.NOT= 000002          A.PRM   000012              IP.ABO= 000040          MC$BID  001765          MC$SFL  002044
AF.OOB= 000004          A.PROC  000020              IP.FAK= 000020          MC$BMC  001767          MC$SYC  000012
AF.OVR= 000002          A.RECE  000016              IP.PND= 000100          MC$BSC  001766          MC$SYL  000036
AF.QUE= 000100          A.REL   000000              IP.UMR= 000200          MC$CAC  001442          MC$SYR  000050
AF.XCC= 000001          A.RES = 000032              I$$RAR= 000000          MC$CAP  001440          MC$SYS  000024
AK.BUF= 000200          A.SBUF  000022              I$$RDN= 000000          MC$CCL  001445          MC$SYZ  000000
AK.GBI= 000202          A.SLEN  000024              I.AST   000022          MC$CDC  002045          MC$TBR  001140
AK.GGF= 000303          A.SMAP  000020              I.ATTL= 000044          MC$CDP  001441          MC$TBS  001141
AK.OCB= 000201          A.STA   000015              I.EFN   000003          MC$CIF  001465          MC$TDR  002053
ALCHOP= ****** GX       A.TRAN  000022              I.FCN   000012          MC$CLD  001464          MC$TIM  000000
AS.CAA= 000006          BYETBL  000000R         002 I.IOSB  000014          MC$COU  100000          MC$TYP  007777
AS.DIS= 000002          CM.CDS= 000004              I.LGTH= 000044          MC$CTL  001454          MC$UBU  002052
AS.DLT= 000001          CM.CEN= 000003              I.LNK   000000          MC$CTR  001452          MC$UFD  002047
AS.FPA= 000001          CM.ELM= 000005              I.LN2   000006          MC$CTS  001453          MC$UMR  001132
AS.PFA= 000004          CM.EXT= 000006              I.PRI   000002          MC$DOV  002050          MC$UMS  001133
AS.RCA= 000002          CM.IND= 000002              I.PRM   000024          MC$LBR  001750          MC$VAX  000003
AS.REA= 000005          CM.INE= 000001              I.TCB   000004          MC$LBS  001751          MC$WID  060000
AS.RRA= 000003          CM.LKT= 000007              I.UCB   000010          MC$LDI  001774          MC$WIH  040000
A$ALI   000012          CM.MSG= 000011              I.XDBF  000040          MC$LDO  001775          MC$WIL  020000
A$BUF   000006          CM.RMT= 000010              I.XDBL  000044          MC$LDR  001762          MC$W08  020000
A$DES   000022          CP.DSB= 000010              I.XIOP  000002          MC$LDS  001763          MC$W16  040000
A$FLG   000000          CP.EXT= 000400              I.XLEN= 000046          MC$LLB  002021          MC$W32  060000
A$LEN   000103          CP.LGO= 000004              I.XLNK  000000          MC$LLP  002115          MC$XBR  001750
A$NUM   000010          CP.MSG= 000002              I.XMOD  000006          MC$LLR  002007          MC$XBS  001751
A$OPT   000002          CP.NIO= 000100              I.XPBF  000022          MC$LOO  000001          MC$XCJ  000322
A$PAR   000004          CP.NUL= 000001              I.XPBL  000026          MC$LRB  002020          MC$XCR  002260
A$SCO   000020          CP.PRV= 000020              I.XPBV  000030          MC$LRP  002114          MC$XCS  002261
A$TYP   000011          CP.RST= 000200              I.XRBF  000012          MC$LRR  002006          MC$XDR  001762
A$$CHK= 000000          CP.SGL= 000040              I.XRBL  000016          MC$LSI  002032          MC$XDS  001763
A$$CPS= 000000          C$$CKP= 000000              I.XTCB  000004          MC$LST  002033          MC$XFR  002272
A$$PRI= 000000          C$$ORE= 000400              I.XTMO  000020          MC$LUP  006072          MC$XFS  002273
A$$TRP= 000000          C$$RSH= 177564              I.XTTB  000032          MC$MAP  010000          MC$XLJ  000323
A.ACC   000014          C.PCPL  000011              I.XTTL  000036          MC$MBL  001764          MC$XLR  002330
A.ACCE  000000          C.PDPL  000010              K$$CNT= 177546          MC$MBX  004420          MC$XMA  000310
A.AST   000006          C.PNAM  000002              K$$CSR= 177546          MC$MBY  001752          MC$XMC  002305
A.BYT   000004          C.PRMT  000012              K$$LDC= 000000          MC$MPX  004406          MC$XMS  002304
A.CALL  000024          C.PSTS  000006              K$$TPS= 000074          MC$NAP  001604          MC$XNR  002332
A.CBL   000002          C.PTCB  000000              LC$EXT  000000          MC$NBR  001130          MC$XRC  002316
A.CONN  000012          DECBYE  000040R             LC$INT  000001          MC$NBS  001131          MC$XRR  002331
A.DEQU  000002          D$$BUG= 177514              LD$LP = 000000          MC$NCR  001154          MC$XRS  002342
A.DIS   000002          D$$ISK= 000000              LOGOUT  000000RG        MC$NCS  001155          MC$020  000002
A.DISC  000014          D$$L11= 000001              L$$ASG= 000000          MC$NML  001274          MD$CI   000007
A.DQSR  177776          D$$YNC= 000000              L$$DRV= 000000          MC$NMR  001142          MD$CNA  000003
A.FLEN  000011          D$$YNM= 000000              L$$P11= 000001          MC$NMS  001143          MD$COU  000001
A.IBUF  000014          E$$XPR= 000000              L$$11R= 000000          MC$NNO  001606          MD$DA   000010
A.ILEN  000016          F$$LVL= 000001              MAXACC= 000020          MC$NNU  001605          MD$DL   000004
A.IMAP  000012          GS.DEL= 000001              MAXID = 000020          MC$NOP  001607          MD$DLV  000020
A.INPU  000006          G$$TPP= 000000              MAXPAS= 000010          MC$NOR  000000          MD$DMC  000014
A.IOS   000026          G$$TSS= 000000              MB$DIS  000001          MC$NPF  001616          MD$DMF  000046
A.KSR5  177774          G$$TTK= 000000              MB$ENA  000000          MC$NPR  001630          MD$DMP  000022
```

```
                                                          F  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-2
Symbol table
```

```
                                                            E 9
MANFRK - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:30  Page 7-3
Symbol table

T3.RST= 000400          UA.CAL= 000100          UA.SPE= 000020          UPDCTX= ****** GX       $SAVAL= ****** GX
T3.SLV= 002000          UA.COM= 000200          UA.TRA= 002000          V$$CTR= 001000          $$$   = 000062
T3.SWS= 000002          UA.ECH= 000004          UA.TRN= 001000          WBUFF = ****** GX       .NAUTO= ****** GX
UA.ACC= 000001          UA.PRO= 000002          UA.TYP= 000010          X$$DBT= 000000          .NSTBL= ****** GX
UA.ALL= 000400          UA.PUT= 000040

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000360    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 15566 Words  ( 61 Pages)
Size of core pool: 16552 Words  ( 63 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:12.25
SY:MANFRK.V2,[135,134]MANFRK/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANFRK
```

```
                                                            F 9
MANFRK      CREATED BY  MACRO  ON 29-JUN-85 AT 10:30     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
   368 000624  012716  000002           MOV     #2,(SP)                 ; IF ADDRESS, LENGTH=2 (PLUS 0 BYTE)
   369 000630  100001           20$:    BPL     30$                     ; IF PL WE HAVE BASIC LENGTH
   370 000632  005016                   CLR     (SP)                    ; ELSE, IF MI - LENGTH=0 (PLUS MI BYTE)
   371 000634  005216           30$:    INC     (SP)                    ; INCLUDE FORMAT BYTE IN LENGTH
   372 000636  000424                   BR      100$
   373                          ;
   374                          ; OBJECT INITIALIZE
   375                          ;
   376 000640  012704           OBCHIN::MOV     (PC)+,R4
   377 000642     007                   .BYTE   ME$OBJ
   378 000643     223                   .BYTE   MF$CHA!CM$RSX
   379 000644  000402                   BR      OBID
   380 000646  012704           OBREIN::MOV     (PC)+,R4
   381 000650     007                   .BYTE   ME$OBJ
   382 000651     224                   .BYTE   MF$REA!CM$RSX
   383 000652                   ARID:
   384 000652  012746  000001   OBID:   MOV     #1,-(SP)                ; ASSUME SINGLE BYTE LENGTH
   385 000656  105711                   TSTB    (R1)                    ; KNOWN OBJECTS ?
   386 000660  001001                   BNE     40$                     ; IF NE, YES - 1 BYTE
   387 000662  005216                   INC     (SP)                    ; ELSE, 2 BYTES
   388 000664  000411           40$:    BR      100$                    ; ... ON TO COMMON CODE
   389                          ;
   390                          ; SYSTEM INITIALIZE
   391                          ;
   392 000666  012704           SYCHIN::MOV     (PC)+,R4
   393 000670     006                   .BYTE   ME$SYS
   394 000671     223                   .BYTE   MF$CHA!CM$RSX
   395 000672  000405                   BR      SYID
   396 000674  012704           SYREIN::MOV     (PC)+,R4
   397 000676     006                   .BYTE   ME$SYS
   398 000677     224                   .BYTE   MF$REA!CM$RSX
   399 000700  000402                   BR      SYID
   400 000702  012704           SYZEIN::MOV     (PC)+,R4
   401 000704     006                   .BYTE   ME$SYS
   402 000705     225                   .BYTE   MF$ZER!CM$RSX
   403 000706  005046           SYID:   CLR     -(SP)                   ; ID LENGTH=0 FOR SYSTEM
   404
   405 000710  150004           100$:   BISB    R0,R4                   ; INCLUDE OPTION
   406 000712  000304                   SWAB    R4                      ; GET FUNCTION IN LOW BYTE
   407 000714  112600                   MOVB    (SP)+,R0                ; GET LENGTH OF ID STRING
   408 000716                           CALLR   INIQIO                  ; AND ISSUE QIO
   409
   410                                  .DSABL  LSB
   411                          ;
   412                          ; OPERATE ROUTINE ENTRY POINTS
   413                          ;
   414 000722                   ALCHOP::
   415 000722                   ALREOP::
   416 000722                   ARREOP::
   417 000722                   CICHOP::
   418 000722                   CIREOP::
   419 000722                   CIZEOP::
   420 000722                   LICHOP::
   421 000722                   LIREOP::
   422 000722                   LIZEOP::
   423 000722                   LOCHOP::
   424 000722                   LOREOP::
     50                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     51                                  ;
     52                                  ; MACRO LIBRARY CALLS
     53                                  ;
     54                                          .MCALL  MANDF$,ANBDF$,SAVMAP,RESMAP,RETC
     55
     56 000000                                   MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     57 000000                                   ANBDF$                          ; DEFINE ALIAS NAME BLOCK OFFSETS
     58
     59                                          .IF DF M$$MGE
     60         000002                           BIAS = 2                        ; STORAGE ON STACK FOR MAPPING BIAS
     61                                          .IFF
     62                                          BIAS = 0
     63                                          .ENDC
```

```
                .TITLE  MANVCD - VOLATILE MANAGEMENT ACP SPECIFIC CODE
                .IDENT  /V05.00/
;
; COPYRIGHT (C) 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       MANVCD - CODE SPECIFIC TO THE VOLATILE MANAGEMENT ACP
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                .SBTTL  FTABLE - FUNCTION DISPATCH TABE
        ;+
        ; **-FTABLE-FUNCTION DISPATCH TABLE FOR VOLATILE ACP
        ;
        ; THIS TABLE DEFINES ALL THE FUNCTIONS SUPPORTED BY THE VOLATILE NETWORK
        ; MANAGEMENT ACP.
        ;
        ; ENTRY FORMAT IS:
        ;                       +-------------+-------------+
        ;                       ! Options     ! Function    !
        ;                       +-------------+-------------+
        ;                       ! Unused      ! Flags       !
        ;                       +-------------+-------------+
        ;                       ! Initialize routine        !
        ;                       +---------------------------+
        ;                       ! Operate routine           !
        ;                       +---------------------------+
        ;
        ;-
000000          .PSECT  DATA$   D,RW
000000  FTABLE::
        ;
        ; AREA OPERATIONS
        ;
000000  MNFUN$  MF$REA,ME$ARE,ARREIN,ARREOP                             ; SHOW AREA

                .IF NDF S$$BAS
        ;
        ; ALIAS OPERATIONS
        ;
000010  MNFUN$  MF$CHA!CM$RSX,MO$SET!ME$ALI,ALCHIN,ALCHOP,RF$PRV ; SET ALIAS
000020  MNFUN$  MF$CHA!CM$RSX,MO$CLE!ME$ALI,ALCHIN,ALCHOP,RF$PRV ; CLEAR ALIAS
000030  MNFUN$  MF$REA!CM$RSX,0!ME$ALI,ALREIN,ALREOP             ; SHOW ALIAS
                .ENDC
        ;
        ; CIRCUIT OPERATIONS
        ;
000040  MNFUN$  MF$CHA,MO$SET!ME$CIR,CICHIN,CICHOP,RF$PRV       ; SET CIRCUIT
000050  MNFUN$  MF$CHA,MO$CLE!ME$CIR,CICHIN,CICHOP,RF$PRV       ; CLEAR CIRCUIT
000060  MNFUN$  MF$ZER,MO$ZER!ME$CIR,CIZEIN,CIZEOP,RF$PRV       ; ZERO CIRCUIT
000070  MNFUN$  MF$REA,ME$CIR,CIREIN,CIREOP                     ; SHOW CIRCUIT
        ;
        ; LINE OPERATIONS
        ;
000100  MNFUN$  MF$CHA,MO$SET!ME$LIN,LICHIN,LICHOP,RF$PRV       ; SET LINE
000110  MNFUN$  MF$CHA,MO$CLE!ME$LIN,LICHIN,LICHOP,RF$PRV       ; CLEAR LINE
000120  MNFUN$  MF$ZER,MO$ZER!ME$LIN,LIZEIN,LIZEOP,RF$PRV       ; ZERO LINE
000130  MNFUN$  MF$REA,ME$LIN,LIREIN,LIREOP                     ; SHOW LINE
        ;
        ; LOGGING OPERATIONS
        ;
000140  MNFUN$  MF$CHA,MO$SET!ME$LOG,LOCHIN,LOCHOP,RF$PRV       ; SET LOGGING
000150  MNFUN$  MF$CHA,MO$CLE!ME$LOG,LOCHIN,LOCHOP,RF$PRV       ; CLEAR LOGGING
000160  MNFUN$  MF$REA,ME$LOG,LOREIN,LOREOP                     ; SHOW LOGGING

                .IF NDF R$$PRO  ; PRO/DECNET

        ;
```

```
                                                .SBTTL  PSIBYE - LOGOUT cleanup for PSI user
                                        ;+
                                        ; *** PSIBYE - LOGOUT cleanup for PSI user
                                        ;
                                        ; INPUTS
                                        ;       $PSIPT - Address of PSI homeblock
                                        ;       $TKTCB - Current TCB address
                                        ;
                                        ; OUTPUTS
                                        ;       R3 - R5 preserved
                                        ;       Remote DTE descriptors for this terminal are deallocated.
                                        ;
                                        ;-

000000                                  PSIBYE::
000000                                          SWSTK$  100$
000004                                          SAVMAP                          ;; Save current mapping

000010  017700  000000G                         MOV     @PSIPT,R0               ;; Get address of PSI home block; RJK01
000014  001433                                  BEQ     50$                     ;; Branch if no PSI support     ;**-1
000016  062700  000004                          ADD     #H$RDTE,R0              ;; Point to remote DTE list
000022  010005                                  MOV     R0,R5                   ;; Fake a mapped address too
000024  017702  000000G                         MOV     @TKTCB,R2               ;; Get address of current TCB   ; RJK01
                                                                                                                ;**-1
000030  010003                          10$:    MOV     R0,R3                   ;; Make R3 address of 'previous' RDT
000032  011500                                  MOV     (R5),R0                 ;; Get unmapped address of 'next' RDT
000034  001423                                  BEQ     50$                     ;; Branch if list exhausted
000036                                          CEACC$  R0,R5                   ;; Map to new RDT
000046  026265  000026  000010                  CMP     T.UCB(R2),R$OWN(R5)     ;; Does this RDT belong to this TI:?
000054  001365                                  BNE     10$                     ;; Branch if not owned here

000056  010346                                  MOV     R3,-(SP)                ;; Address of previous block
000060  010046                                  MOV     R0,-(SP)                ;; Address of current block
000062                                          CALLX   $XLINK,AUX              ;; Unlink control block
000072  012701  000024                          MOV     #R$LEN,R1               ;; Get length of block to deallocate
000076                                          CALL    DEACB                   ;; Deallocate the block
000102  000752                                  BR      10$                     ;; Loop until all blocks gone

000104                                  50$:    RESMAP                          ;; Restore mapping
000110                                  100$:   RETURN

        000001                                  .END
```

```
                                        .SBTTL  CHKACC - CHECK ACCESS CONTROL FIELDS
                         ;+
                         ; **-CHKACC-CHECK ACCESS CONTROL FIELDS
                         ;
                         ; THIS ROUTINE CHECKS THE ACCESS CONTROL FIELDS FOR A DESINATION,
                         ; AND IF VALID MOVES THEM INTO THE SPECIFIED BUFFER.
                         ;
                         ; INPUTS:
                         ;       R0 = TARGET BUFFER ADDRESS
                         ;       R2 = BYTES REMAINING IN INPUT BUFFER
                         ;       R3 = INPUT BUFFER
                         ;       R5 = MAXIMUM LENGTH OF FIELD BEING CHECKED
                         ;
                         ; OUTPUTS:
                         ;       IF C-CLEAR, SUCCESS
                         ;          R0 = NEXT BYTE IN TARGET BUFFER
                         ;          R2 = BYTES REMAINING IN INPUT BUFFER
                         ;          R3 = NEXT BYTE IN INPUT BUFFER
                         ;       IF C-SET, ERROR
                         ;
                         ; REGISTERS:
                         ;       R4 IS PRESERVED
                         ;
                         ;-
000470                   CHKACC:
000470  121305                   CMPB    (R3),R5                 ; IS THE FIELD TOO LONG ?
000472  003013                   BGT     20$                     ; IF GT, YES - ERROR
000474  112305                   MOVB    (R3)+,R5                ; GET LENGTH OF FIELD
000476  005302                   DEC     R2                      ; DEC BUFF COUNT FOR LENGTH BYTE
000500  002410                   BLT     20$                     ; IF LT, BUFFER LENGTH ERROR
000502  000241                   CLC                             ; ASSUME SUCCESS
000504  110520                   MOVB    R5,(R0)+                ; SET LENGTH OF STRING IN BUFFER
                         ;
                         ; MOVE STRING INTO BUFFER
                         ;
000506  005305           10$:    DEC     R5                      ; DEC STRING LENGTH COUNT
000510  002405                   BLT     30$                     ; IF LT, FINISHED
000512  005302                   DEC     R2                      ; DEC BUFFER COUNT
000514  002402                   BLT     20$                     ; IF LT, BUFFER LENGTH ERROR
000516  112320                   MOVB    (R3)+,(R0)+             ; MOVE CHAR INTO BUFFER
000520  000772                   BR      10$                     ; AND LOOP

000522  000261           20$:    SEC
000524                   30$:    RETURN

        000001                   .END
```

```
000230  062701  000000                   ADD     #D$ANN,R1               ;; POINT TO ALIAS LISTHEAD
000234  010167  000000G                  MOV     R1,$CRUNM               ;; SET UP CURRENT AND PREV UNMAP PTRS
000240  016767  000000G 000000G 60$:     MOV     $CRUNM,$PRUNM           ;; ...
000246  011167  000000G                  MOV     (R1),$CRUNM             ;; GET NEW CURRENT POINTER
000252  011146                           MOV     (R1),-(SP)              ;; SET UP TO MAP TO ANB
000254                                   CALL    @CEACC                  ;; MAP TO ANB
000260  012601                           MOV     (SP)+,R1                ;; RECOVER VIRTUAL ADDRESS TO ANB
000262  001406                           BEQ     70$                     ;; IF EQ, AT END OF LIST - LINK HERE
000264  005761  000010                   TST     A.UCB(R1)               ;; IS THIS BLOCK GLOBAL ?
000270  001012                           BNE     80$                     ;; IF NO, CHECK FOR ORDER
000272  005764  000020          65$:     TST     A$SCO(R4)               ;; ELSE, IS NEW ONE GLOBAL ALSO ?
000276  001760                           BEQ     60$                     ;; IF EQ, YES - FORCE TO END OF GLOBALS
000300                          70$:     ; LINK BLOCKS
                                         ; (SP) = UNMAPPED ADDRESS OF NEW BLOCK (ALREADY ON STACK)
                                         ; FRUNM = UNMAPPED ADDRESS OF PREVIOUS BLOCK
000300  016746  000000G                  MOV     $PRUNM,-(SP)            ;; PREVIOUS UNMAPPED ADDRESS
000304                                   CALLX   $XLINK,AUX              ;; LINK NEW ANB IN LIST
000314  000405                           BR      90$                     ;; AND RETURN

000316  026164  000010  000020  30$:     CMP     A.UCB(R1),A$SCO(R4)     ;; DOES THIS ANB BELONG BEFORE OUR'S ?
000324  101745                           BLOS    60$                     ;; IF LOS, YES - CONTINUE SCAN
000326  000761                           BR      65$                     ;; ELSE CHECK FOR GLOBAL SCOPE

000330                          90$:     RESMAP                          ;; RESTORE MAPPING

                                         .IF DF R$$PRO                   ;; PRO/DECNET

                                         MOV     @DECPT,R1               ;; POINT TO DECNET HOME BLOCK
                                         BIC     #DF.AUP,D$STAT(R1)      ;; UNLOCK NT.CFG PARTITION

                                         .ENDC

000334                          100$:    RETURN                          ;; AND BACK TO USER STATE AND RETURN
```

```
                                        .SBTTL  PWRAST  - Powerfail Recovery AST
                                        .SBTTL  ABOAST  - Abort Cleanup AST
                                ;+
                                ;                               *** PWRAST/ABOAST ***
                                ;
                                ; Function:     To perform any termination and cleanup of in-progress or queued
                                ;               load requests at the time of a system powerfail or a task abort
                                ;               from a privileged terminal.
                                ;
                                ; Call Format:  Not applicable
                                ;
                                ; Inputs:       No explicit inputs
                                ;
                                ; Outputs:      Current load sequence ended,  load request listhead in AUX is
                                ;               emptied with a "halt load" dispatch for each CCB dequeued.
                                ;
                                ; Registers:    R0 through R5 destroyed
                                ;
                                ; Notes:        When RSX systems recover from powerfail,  any active tasks
                                ;               with powerfail ASTs are dispatched to before any device
                                ;               drivers are dispatched to at their powerfail recovery
                                ;               entry points.
                                ;
                                ;               MLD stays in system state until all load requests have been
                                ;               terminated.  This is so that AUX will not queue more CCBs
                                ;               while MLD is trying to clean up and exit.
                                ;-

 000754                         PWRAST:
 000754                         ABOAST: .IFT                                    ;; If system is RSX-11M ...

                                        CLOSE$  #FDB                            ;; Close the microcode file

                                        .IFTF                                   ;; For any operating system ...

 000754                                 CALL    $SWSTK,40$                      ;; Switch to system state

                                        .IF     DF      N$$UMR                  ;; If system has UMRs ...

 000760 016704  000144'                 MOV     LDBPTR,R4                       ;; Get address of CCB/LDB
 000764 001402                          BEQ     5$                              ;; Branch if none
 000766                                 CALL    @LDBRT                          ;; Return the CCB and LDB to pool

 000772                         5$:     .ENDC                                   ;; End UMR conditional

 000772 016704  000156'                 MOV     CURQST,R4                       ;; Get address of current CCB
 000776 001010                          BNE     20$                             ;; Branch if none

 001000 017700  000602'         10$:    MOV     @TKTCB,R0                       ;; Get our TCB address
 001004 062700  000012                  ADD     #T.RCVL,R0                      ;; Advance to receive listhead
 001010                                 CALL    @QRMVF                          ;; Dequeue a load request
 001014 103421                          BCS     30$                             ;; Branch if none left
 001016 010104                          MOV     R1,R4                           ;; Copy CCB address

 001020                         20$:    CALL    @STDDM                          ;; Get PDV index and line table address
 001024 010403                          MOV     R4,R3                           ;; Copy CCB address
 001026 062703  000012                  ADD     #C.STS,R3                       ;; Advance to status field
```

```
                                                          F  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

ABOAST    000754 R       9-319     #12-649
AC$DNT  = 000002        #6-83
AC$X25  = 000001        #6-83
AE$CIR  = 000003        #6-83
AE$LIN  = 000001        #6-83
AE$MOD  = 000004        #6-83
BLOCK     000152 R      #8-210     *16-881    *16-882     18-954     18-974     18-982
BLXIO     000572 RG     #8-230      10-529
CCBRT     000604 RG     #8-238      9-355      12-682
CF.LB   = 100000         18-978
CL$ASZ  = 010500        #6-83       7-143
CL$DLL  = 000500        #6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83
                         6-83       6-83
CL$ECL  = 000300        #6-83       6-83
CL$LDN  = 010400        #6-83       6-83
CL$MAN  = 000000        #6-83       6-83       6-83       6-83
CL$PAZ  = 034100        #6-83       6-83       6-83
CL$PLH  = 034000        #6-83       6-83       6-83       6-83       6-83
CL$PLL  = 000600        #6-83
CL$PRT  = 034200        #6-83
CL$ROU  = 010000        #6-83       6-83       6-83
CL$SES  = 000200        #6-83       6-83       6-83
CL$SGE  = 035000        #6-83       6-83       6-83
CL$SSE  = 035100        #6-83       6-83       6-83       6-83       6-83
CL$TRN  = 000400        #6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83
                         6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83       6-83
                         6-83
CL$XL2  = 013700        #6-83
CL$XL3  = 013600        #6-83       6-83
CL$X2S  = 013500        #6-83       6-83
CS.ERR  = 100000         9-464      10-571     12-677
CS.FTL  = 001000         9-444      12-677
CTLNUM    000162 R      #8-213     *9-405      11-600
CURQST    000156 R      #8-211     *9-310      9-353     *9-356     *9-363      9-388      9-442      9-462      12-665
                        *12-686     18-979
C.BUF1    000014         9-333      9-334      9-429      10-518     10-519    *10-532    *10-558     18-955
C.CNT1    000020         10-521    *10-531    *10-559
C.FNC     000010        *9-335      9-364     *9-447     *9-463      10-546
C.LIN     000006         9-390     *9-395
C.MOD     000011         10-508     10-550
C.STS     000012        *9-465      10-556     12-676
DDMTBL    000164 R      #8-214     *9-404      10-544
DEVERR    000051 R      #7-148      10-573
DL$AST  = 000002        #6-83
DL$HLT  = 000000        #6-83
DL$IST  = 000001        #6-83
DL$MAI  = 000004        #6-83
DL$OFF  = 000001        #6-83
DL$ON   = 000000        #6-83
DL$RUN  = 000003        #6-83
DL$SHU  = 000002        #6-83
DL$SYN  = 000005        #6-83
```

```
                                                          G  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
     44                                                 .SBTTL  MACRO CALLS AND DATA
     45                                         ;
     46                                         ; MACRO CALLS
     47                                         ;
     48                                                 .MCALL  IOERR$,CALLR,SAVRG,RESRG,PKTDF$,MANDF$,SAVMAP,RESMAP,MAP,CCBDF$
     49
     50 000000                                          IOERR$                          ; DEFINE RSX ERROR CODES
     51 000000                                          PKTDF$  <:>                     ; DEFINE I/O PACKET OFFSETS
     52 000000                                          MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     53 000000                                          CCBDF$                          ; DEFINE COMMUNICATIONS CONTROL BLOCK SYMBOLS
     54
     55                                                 .IF DF R$$PRO   ; DECNET/PRO
     56
     57                                                 .MCALL DHBDF$
     58
     59                                                 DHBDF$                          ; DEFINE DECNET HOME BLOCK OFFSETS
     60
     61                                                 .ENDC   ; DF R$$PRO
     62
     63                                         ;
     64                                         ; LOCAL MACRO DEFINITIONS
     65                                         ;
     66                                                 .MACRO  IOFUN$  FUNCTION,ROUTINE ; I/O FUNCTION DISPATCH TABLE ENTRY
     67                                                 .WORD   FUNCTION/256.
     68                                                 .WORD   ROUTINE
     69                                                 .ENDM   IOFUN$
     70
     71                                         ;
     72                                         ; DEFINITIONS
     73                                         ;
     74                 000000                          MN$VOL=0                        ; ASSEMBLE FOR VOLATILE ACP
     75
     76                 000466                          CTXSIZ==310.                    ; SIZE OF A CONTEXT AREA
     77                 000454                          BUFSIZ=300.                     ; SIZE OF LOCAL BUFFER
     78                 000062                          MAXID=50.                       ; MAXIMUM SIZE OF AN ID
     79                 000020                          MAXLID=16.                      ; MAXIMUM SIZE OF A LINE-ID
     80
     81                 000200                          CM$RSX=200                      ; FLAG RSX SPECIFIC COMMAND
     82
     83 000000                                  NMVVEC::                                ; TABLE OF VECTORS FOR NMVACP
     84 000000          000000                          .WORD   0
     85 000002          000000G                 ISAR5:: .WORD   KISAR5
     86 000004          000000G                 ISAR6:: .WORD   KISAR6
     87 000006          000000G                 KSAR6:: .WORD   KISAR6
     88 000010          000000G                 AALOCB::.WORD   $ALOCB
     89 000012          000000G                 BLXIO:: .WORD   $BLXIO
     90 000014          000000G                 COPT::  .WORD   $COPT
     91 000016          000000G                 DDEACB::.WORD   $DEACB
     92 000020          000000G                 DREXT:: .WORD   $DREXT
     93 000022          000000G                 EXRQN:: .WORD   $EXRQN
     94 000024          000000G                 EXRQP:: .WORD   $EXRQP
     95 000026          000000G                 FMASK:: .WORD   $FMASK
     96 000030          000000G                 HEADR:: .WORD   $HEADR
     97 000032          000000G                 IOFIN:: .WORD   $IOFIN
     98 000034          000000G                 MOULS:: .WORD   $MOULS
     99 000036          000000G                 RELOC:: .WORD   $RELOC
    100 000040          000000G                 SRSTD:: .WORD   $SRSTD
```

```
   629                                                  .SBTTL  MANCLN - PROCESS NM.CLN REQUESTS
   630                                          ;+
   631                                          ; **-MANCLN-PROCESS NM.CLN REQUESTS
   632                                          ;
   633                                          ; THIS ROUTINE DEALLOCATES THE CONTEXT AREA (IF ONE EXISTS).
   634                                          ;
   635                                          ; INPUTS:
   636                                          ;       R3 = ADDRESS OF I/O PACKET
   637                                          ;
   638                                          ; OUTPUTS:
   639                                          ;       R0 = I/O STATUS TO RETURN (IS.SUC)
   640                                          ;       R1 = 0
   641                                          ;
   642                                          ; REGISTERS:
   643                                          ;       R3 IS PRESERVED
   644                                          ;-
   645
   646 002702                                   MANCLN:
   647 002702   016705  175314                          MOV     TSKLN2,R5               ; GET WINDOW BLOCK ADDRESS
   648 002706   001407                                  BEQ     10$                     ; IF EQ, NO WINDOW BLOCK - JUST RETURN
   649 002710   016567  000002  175274                  MOV     W$CTX(R5),CURCTX        ; ELSE, SET CURRENT CONTEXT
   650 002716                                           CALL    UPDCTX                  ; RESTORE THE CONTEXT AREA
   651 002722   005267  175270                          INC     DELFLG                  ; MARK CONTEXT FOR DELETION
   652 002726   012700  000001                  10$:    MOV     #IS.SUC,R0              ; SET STATUS = SUCCESS
   653 002732   005001                                  CLR     R1                      ; ...
   654 002734                                           RETURN
```

```
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 6
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              6-307     6-312     6-399     6-402     6-404     6-412     6-423     6-427     6-432     6-434
                  6-437     6-467     8-568     8-571     8-589     8-591     9-619     9-623     10-650
CALLR            #5-48      6-365     6-366     8-557
CCBDF$           #5-48      5-53
IOERR$           #5-48      5-50
IOFUN$           #5-66      6-472     6-473     6-474     6-475
MANDF$           #5-48      5-52
MAP              #5-48
PKTDF$           #5-48      5-51
RESMAP           #5-48
RESRG            #5-48
RETURN            6-405     7-513     10-654    11-694
SAVMAP           #5-48
SAVRG            #5-48
SWSTK$            6-312     6-427
```

```
**FILE**ID**MANAST
```

```
                        .TITLE  MANBUF - NETWORK MANAGEMENT ACP BUFFER ROUTINES
                        .IDENT  /V05.00/
                ;
                ; COPYRIGHT (C) 1981, 1983, 1984, 1985 BY
                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                ;
                ;
                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                ; TRANSFERRED.
                ;
                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                ; CORPORATION.
                ;
                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                ;
                ;
                ; MODULE DESCRIPTION:
                ;
                ;       NETWORK MANAGEMENT ACP BUFFER ROUTINES
                ;
                ;
                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                ;
                ; IDENT HISTORY:
                ;
                ; 4.00  07-NOV-83
                ;       DECNET-11M V4.0
                ;       DECNET-11M-PLUS V2.0
                ;
                ; 5.00  22-JUL-85
                ;       DECnet-11M/S V4.2
                ;       DECnet-11M-Plus V3.0
                ;       DECnet-Micro/RSX V1.0
                ;
                ;
```

F 7

```
MANBUF      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00
SYMBOL  VALUE            REFERENCES
$SAVAL  = ******  GX     8-264        9-306       10-353      10-358      11-395
```

G 7

```
MANBUF      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 3
MACRO CROSS REFERENCE                                     CREF   04.00
```

```
                                                    F  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-2
Symbol table

MD$DMR   000050     ME$LIN   000001     MF$DST=  000001     MM$OVR=  000020     MP$BLO   001452
MD$DMV   000042     ME$LOG   000002     MF$DUM   000020     MM$PER=  000002     MP$BMX   002176
MD$DN    000016     ME$MOD   000004     MF$EVT   000001     MM$ZRO=  000040     MP$BNP   006330
MD$DP    000000     ME$NOD   000000     MF$FUP   000264     MN$UNL   000377     MP$BRT   001620
MD$DPV   000044     ME$NON   177777     MF$GDT=  000100     MO$ACC   000200     MP$BSA   002176
MD$DQ    000006     ME$OBJ   000007     MF$GNM=  000020     MO$ADD   000002     MP$BSD   002203
MD$DTE   000024     ME$OB2   000004     MF$GRO=  000010     MO$ALA   000004     MP$BSI   002200
MD$DU    000002     ME$PRO   000005     MF$GTY=  000040     MO$ALI   000000     MP$BSP   006331
MD$DUP   000012     ME$SYS   000006     MF$KNO   177777     MO$CHA   000040     MP$BUF   000170
MD$DV    000026     ME$XAC=  000001     MF$LMX=  004000     MO$CIR   000003     MP$BUP   006324
MD$DZ    000030     ME$XPR=  000002     MF$LOA   000017     MO$CLE   000100     MP$BUS   001643
MD$FUL   000000     ME$X5S=  000003     MF$LOD   177775     MO$COU   000060     MP$CAC   001750
MD$HAL   000001     ME$X9S=  000004     MF$NET=  000002     MO$DAC   000000     MP$CAS   001762
MD$HEL   000002     ME.BLD   177744     MF$PCH=  002000     MO$DEF   000000     MP$CAT   002210
MD$KCP   000013     ME.CON   177753     MF$PCT=  000002     MO$DPR   000000     MP$CCS   004406
MD$KDP   000034     ME.CST   177765     MF$PDT=  001000     MO$ENT   000017     MP$CHN   002141
MD$KDZ   000036     ME.DIS   177755     MF$PLN=  004000     MO$EVE   000100     MP$CIR   000144
MD$KL    000040     ME.DON   177600     MF$PMC=  010000     MO$INF   000160     MP$CLK   002131
MD$KMX   000054     ME.FCO   177762     MF$REA   000024     MO$INS   000002     MP$CLN   002126
MD$KMY   000052     ME.FIO   177756     MF$RMX=  010000     MO$LIN   000001     MP$CLT   002211
MD$LEN   000050     ME.FOP   177763     MF$RRT=  002000     MO$LOG   000001     MP$CMB   002142
MD$PCL   000011     ME.FOR   177776     MF$RST=  001000     MO$MIR   000031     MP$CMK   000537
MD$QNA   000005     ME.FUN   177777     MF$SAC=  001000     MO$NAM   000001     MP$CMX   002153
MD$UNA   000001     ME.GRO   177745     MF$SAD=  000400     MO$NIC   000023     MP$CND   000310
MD$WIT   000002     ME.HAR   177750     MF$SGR=  020000     MD$NOD   000000     MP$CNU   001753
MD.ABD   000017     ME.IID   177767     MF$SIG   177773     MO$OFF   000001     MP$COB   000311
MD.ACC   000010     ME.LCO   177766     MF$SMX=  020000     MO$ON    000000     MP$CON   002126
MD.AOB   000016     ME.LPR   177757     MF$SND=  002000     MO$OPT   000001     MP$COS   001604
MD.BOB   000011     ME.MPR   177773     MF$SNU=  000200     MO$PER   000200     MP$CPF   000230
MD.DIA   [illegible]007     ME.MVE   177771     MF$SDB=  000010     MO$PRO   000002     MP$CPL   000232
MD.DOB   0[illegible]15     ME.OPE   177747     MF$SPF   000302     MO$PRS   000001     MP$CPT   000226
MD.DUM   000002     ME.PLO   177751     MF$SPR=  000020     MO$REA   000200     MP$CPU   000161
MD.FAI   000014     ME.PMI   177743     MF$SPS=  004000     MD$SET   000000     MP$CSZ   001755
MD.FNA   000001     ME.PNA   177752     MF$STA=  000001     MO$STA   000020     MP$CUS   002127
MD.FOB   000006     ME.PRI   177775     MF$SUS=  010000     MO$SUM   000000     MP$CVA   000540
MD.LOA   000001     ME.PTY   177772     MF$SYS   000026     MO$TSK   000000     MP$DAL   002570
MD.LSH   000020     ME.PVA   177760     MF$TES   000022     MO$VOL   000000     MP$DCO=  001464
MD.NOB   000012     ME.RES   177761     MF$TRA   000301     MO$ZER   000000     MP$DDT   002177
MD.NON   177777     ME.ROO   177754     MF$TRI   000021     MPVSTA=  005214     MP$DEL   001131
MD.PER   000000     ME.SIZ   177774     MF$WDF=  000040     MP$ACB   000012     MP$DES   000156
MD.RES   000004     ME.SYS   177746     MF$WMX=  000100     MP$ACC   000514     MP$DEV   002114
MD.ROB   000005     ME.UCO   177770     MF$ZER   000025     MP$ACT   004526     MP$DFA   001320
MD.RSH   000013     MF$ACT   177776     ML$ALL   100000     MP$ADD   000766     MP$DHO=  001465
MD.SDU   000005     MF$ADD   000000     ML$CLS   000000     MP$ADJ   001440     MP$DIA   000173
MD.SLO   000003     MF$ADJ   177774     ML$CON   000001     MP$ADP   006325     MP$DLB   002571
MD.SNA   000000     MF$ALL   177775     ML$EXT   000001     MP$ADS   004406     MP$DLG   004521
MD.TLO   000004     MF$ANB=  000010     ML$FIL   000002     MP$ALB   000036     MP$DLI=  001466
MD.UNA   000002     MF$ASC=  000020     ML$FIR   000000     MP$AMC   001640     MP$DLT   002200
MD.UNR   000003     MF$BDF=  000010     ML$INT   000000     MP$AMH   001641     MP$DRO   001441
MD.UOB   000007     MF$BMX=  000020     ML$KNO   140000     MP$ANB   004420     MP$DST   000454
MD.VOL   000006     MF$BYE   000300     ML$MON   000003     MP$ASB   000024     MP$DTE   002140
ME$ALI   000010     MF$CAT=  000200     ML$SYS   000002     MP$ASC   004432     MP$DTY=  001452
ME$AL2   000003     MF$CHA   000023     ML$TOP   000001     MP$AUS   000512     MP$DUA   000207
ME$ARE   000005     MF$CLT=  000400     MM$ALL=  000010     MP$BBT   002165     MP$DUC   000210
ME$CIR   000003     MF$CMK=  000040     MM$DON=  000001     MP$BDF   002164     MP$DUM   000202
ME$EXA   000000     MF$CVA=  000100     MM$KNO=  000004     MP$BFQ   002121     MP$DUP   002127
ME$EXE   000200     MF$DNT=  020000     MM$MUX=  000100     MP$BLK   001616     MP$DVC   002114
```

```
                                                    G  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-3
Symbol table
```

```
                                                                         F  9
MANFRK      CREATED BY  MACRO  ON 29-JUN-85 AT 10:30       PAGE 1
SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL  VALUE             REFERENCES

BUFFER  = ******  GX      6-144       6-146
CF$FDN  = ******  GX      7-181       7-186
CONTEX  = ******  GX      6-131       7-193
CTXHD   = ******  GX      7-176
CTXSIZ  = ******  GX      6-131       7-193
CURCTX  = ******  GX      6-135      *7-188
CURIOP  = ******  GX      6-127       6-142      *7-198      7-214
C$FLAG  = ******  GX      7-181      *7-186
C$SP    = ******  GX     *6-136       7-191
C$STAT  = ******  GX      7-199
FDNCNT  = ******  GX     *7-187
FORK      000000 RG      #6-100
FRKDON    000202 RG      #7-172
GETPKT  = ******  GX      6-149
ISTACK  = ******  GX      6-133       7-190
I.PRM     000024          6-143      *6-147       7-215
LSTCTX  = ******  GX     *6-137
READ    = ******  GX      7-216
SS$LOA  = 020000         #5-66
SS$NDA  = 040000         #5-65
SS$OUT  = 010000         #5-67        6-111
STATUS  = ******  GX     *7-199
S$DISK    000000         #5-63
S$DOWN    000010         #5-71
S$LOAD    000002         #5-68
S$NAME    000014         #5-73
S$NEXT    000012         #5-72        6-113       6-120
S$SIZE    000004         #5-69
S$STAT    000000         #5-64
S$UP      000006         #5-70        6-109       6-114
S$WIND    000020         #5-74
TKTCB   = ******  GX      6-129       7-195
TSKLN2  = ******  GX      6-128      *7-197
T.UCB     000026          6-130      *7-196
UPDCTX  = ******  GX      7-189
WBUFF   = ******  GX      6-148
$SAVAL  = ******  GX      6-101
.NAUTO  = ******  GX      7-207
.NSTBL  = ******  GX      6-106       6-121
```

```
                                                                         G  9
MANFRK      CREATED BY  MACRO  ON 29-JUN-85 AT 10:30       PAGE 2
```

```
    425 000722                        MOCHOP::
    426 000722                        MOREOP::
    427 000722                        MOZEOP::
    428 000722                        NOCHOP::
    429 000722                        NOREOP::
    430 000722                        NOZEOP::
    431 000722                        OBCHOP::
    432 000722                        OBREOP::
    433 000722                        PRCHOP::
    434 000722                        PRREOP::
    435 000722                        SYCHOP::
    436 000722                        SYREOP::
    437 000722                        SYZEOP::
    438 000722                                CALLR   OPRQIO                  ; ISSUE OPERATE QIO
    439
    440          000001                         .END
```

```
                                        .SBTTL  TRAADD - TRANSLATE AN ADDRESS TO A NAME
                                ;+
                                ; **-TRAADD-TRANSLATE AN ADDRESS TO A NAME
                                ;
                                ; THIS ROUTINE TRANSLATES AN ADDRESS TO A NAME
                                ;
                                ; INPUTS:
                                ;       R1 = POINTER TO ADDRESS
                                ;       R3 = ADDRESS OF BUFFER
                                ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
                                ;
                                ; OUTPUTS:
                                ;       R0 = RETURN STATUS
                                ;       R2 = LENGTH OF DATA TO RETURN (0 OR 6)
                                ;
                                ;-

000000                          TRAADD::
000000  005002                          CLR     R2                      ; ASSUME NO NAME TO RETURN
000002  016705  000000G                 MOV     CURIOP,R5               ; GET ADDRESS OF I/O PACKET
000006  026527  000006G 000002          CMP     I.PRM+6(R5),#2          ; WAS A TWO BYTE ADDRESS SUPPLIED ?
000014  001403                          BEQ     10$                     ; IF EQ, YES - TRY TO MAP IT
000016  012700  177776                  MOV     #ME.FOR,R0              ; ELSE, MESSAGE FORMAT ERROR
000022  000414                          BR      30$                     ; AND RETURN
000024  011101                  10$:    MOV     (R1),R1                 ; GET THE ADDRESS
000026  010300                          MOV     R3,R0                   ; POINT TO NAME BUFFER
000030                                  CALL    MAPADD                  ; MAP THE ADDRESS TO A NAME
000034  103003                          BCC     20$                     ; IF CC, SUCCESS
000036  012700  177770                  MOV     #ME.UCO,R0              ; ELSE ``UNKNOWN COMPONENT''
000042  000404                          BR      30$                     ; ...
000044  012700  000001          20$:    MOV     #MS.SUC,R0              ; SUCCESS !
000050  012702  000006                  MOV     #6,R2                   ; SIX BYTES TO RETURN
000054                          30$:    RETURN
```

```
                                                .SBTTL  MACRO CALLS AND IMPURE DATA
                                        ;
                                        ; MACRO LIBRARY CALLS
                                        ;
                                                .MCALL  MANDF$,ALUN$S,RETC,PDVDF$,GIN$,DIR$

 000000                                         MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
 000000                                         PDVDF$                          ; DEFINE PDV SYMBOLS

          000200                                CM$RSX=200                      ; RSX SPECIFIC MANAGEMENT FLAG
          020000                                FE.CEX=20000                    ; COMM EXEC LOADED FLAG IN $FMASK

                                                .IF     DF      R$$MPL
                                                  .IF   NDF     R$$PRO

                                                .GLOBL  GI.VEC

                                        GETVEC: GIN$    GI.VEC,NMVVEC,NMVVCL    ; DIRECTIVE FORM FOR VECTOR TRANSLATION
                                                                                ; TABLE IS IN THE MANACP MODULE
                                                  .ENDC
                                                .ENDC

 000000                                         .PSECT
```

```
                         ; MODULE OPERATIONS
                         ;
000170                   MNFUN$  MF$CHA,MO$SET!ME$MOD,MOCHIN,MOCHOP,RF$PRV          ; SET MODULE
000200                   MNFUN$  MF$CHA,MO$CLE!ME$MOD,MOCHIN,MOCHOP,RF$PRV          ; CLEAR MODULE
000210                   MNFUN$  MF$ZER,MO$ZER!ME$MOD,MOZEIN,MOREOP,RF$PRV          ; ZERO MODULE
000220                   MNFUN$  MF$REA,ME$MOD,MOREIN,MOREOP                        ; SHOW MODULE

                                 .ENDC   ; NDF R$$PRO

                         ;
                         ; NODE OPERATIONS
                         ;
000230                   MNFUN$  MF$CHA,MO$SET!ME$NOD,NOCHIN,NOCHOP,RF$PRV          ; SET NODE
000240                   MNFUN$  MF$CHA,MO$CLE!ME$NOD,NOCHIN,NOCHOP,RF$PRV          ; CLEAR NODE
000250                   MNFUN$  MF$ZER,MO$ZER!ME$NOD,NOZEIN,NOZEOP,RF$PRV          ; ZERO NODE
000260                   MNFUN$  MF$REA,ME$NOD,NOREIN,NOREOP                        ; SHOW NODE
                                 .IF NDF S$$BAS
                         ;
                         ; OBJECT OPERATIONS
                         ;
000270                   MNFUN$  MF$CHA!CM$RSX,MO$SET!ME$OBJ,OBCHIN,OBCHOP,RF$PRV ; SET OBJECT
000300                   MNFUN$  MF$CHA!CM$RSX,MO$CLE!ME$OBJ,OBCHIN,OBCHOP,RF$PRV ; CLEAR OBJECT
000310                   MNFUN$  MF$REA!CM$RSX,ME$OBJ,OBREIN,OBREOP                ; SHOW OBJECT
                         ;
                         ; PROCESS OPERATIONS
                         ;
                                 .IF NDF R$$RTR
000320                   MNFUN$  MF$CHA!CM$RSX,MO$SET!ME$PRO,PRCHIN,PRCHOP,RF$PRV ; SET PROCESS
000330                   MNFUN$  MF$CHA!CM$RSX,MO$CLE!ME$PRO,PRCHIN,PRCHOP,RF$PRV ; CLEAR PROCESS
                                 .ENDC   ; NDF R$$RTR
000340                   MNFUN$  MF$REA!CM$RSX,ME$PRO,PRREIN,PRREOP                ; SHOW PROCESS
                         ;
                         ; SYSTEM OPERATIONS
                         ;
                                 .IF NDF R$$RTR
000350                   MNFUN$  MF$CHA!CM$RSX,MO$SET!ME$SYS,SYCHIN,SYCHOP,RF$PRV ; SET SYSTEM
000360                   MNFUN$  MF$CHA!CM$RSX,MO$CLE!ME$SYS,SYCHIN,SYCHOP,RF$PRV ; CLEAR SYSTEM
                                 .ENDC   ; NDF R$$RTR

                                 .IF NDF R$$PRO  ; PRO/DECNET

000370                   MNFUN$  MF$ZER!CM$RSX,MO$ZER!ME$SYS,SYZEIN,SYZEOP,RF$PRV ; ZERO SYSTEM
000400                   MNFUN$  MF$REA!CM$RSX,ME$SYS,SYREIN,SYREOP                ; SHOW SYSTEM

                                 .ENDC   ; NDF R$$PRO

                                 .ENDC   ; NDF S$$BAS
                         ;
                         ; SPECIAL FUNCTIONS
                         ;
000410                   MNFUN$  MF$TRA,MO$ALI,TRAALI,0,RF$NCX                      ; TRANSLATE ALIAS
000420                   MNFUN$  MF$TRA,MO$NAM,TRANAM,0,RF$NCX                      ; TRANSLATE NAME
000430                   MNFUN$  MF$TRA,MO$ADD,TRAADD,0,RF$NCX                      ; TRANSLATE ADDRESS
000440                   MNFUN$  MF$TRA,MO$ALA,TRAACC,0,RF$NCX!RF$PRV               ; TRANSLATE ALIAS W/
                                                                                    ;     ACCESS CONTROL
                                 .IF NDF S$$BAS & R$$RTR
000450                   MNFUN$  MF$BYE,0,LOGOUT,RF$NCX                             ; LOGOUT FOR USER
```

```
                                                F 14
PSIBYE - Perform logout cleanup MACRO V05.03b Saturday 29-Jun-85 10:32  Page 6-1
Symbol table

A$$CHK= 000000                   H$LBDA  000070                   M$$OVR= 000000                   TS.OUT= 000400                   T2.AST= 100000
A$$CPS= 000000                   H$LBDN  000072                   N$$ACC= 000001                   TS.RDN= 040000                   T2.CAF= 000400
A$$PRI= 000000                   H$LDTE  000002                   N$$BUF= 000001                   TS.RUN= 004000                   T2.CHK= 020000
A$$TRP= 000000                   H$LEN   000042                   N$$LDV= 000001                   TS.STP= 001000                   T2.CKD= 010000
C$$CKP= 000000                   H$LOTS  000032                   N$$MCP= 000001                   T$$KMG= 000000                   T2.DST= 040000
C$$ORE= 000400                   H$NETW  000024                   N$$MLL= 000001                   T$$MIN= 000000                   T2.FXD= 002000
C$$RSH= 177564                   H$NML = 000006                   N$$MOV= 000010                   T.ACTL  000052                   T2.HLT= 000200
DEACB = ****** GX                H$NPT   000022                   N$$NCT= 000001                   T.ASTL  000016                   T2.REX= 001000
D$$BUG= 177514                   H$PTB   000020                   N$$PEM= 000001                   T.ATT   000062                   T2.SEF= 004000
D$$ISK= 000000                   H$PVC   000006                   PSIBYE  000000RG                 T.CPCB  000004                   T2.SPN= 000004
D$$L11= 000001                   H$RDTE  000004                   PSIPT = ****** GX                T.DPRI  000040                   T2.STP= 000020
D$$YNC= 000000                   H$RNW   000042                   P$$P45= 000000                   T.EFLG  000022                   T2.WFR= 000001
D$$YNM= 000000                   H$SVC   000036                   P$$WRD= 000000                   T.IOC   000003                   T3.ACP= 100000
E$$XPR= 000000                   H$TRB   000016                   Q$$OPT= 000010                   T.LBN   000041                   T3.CAL= 000100
F$$LVL= 000001                   H$XAVL  000100                   R$DAL   000013                   T.LDV   000044                   T3.CLI= 001000
G$$TPP= 000000                   H$XBIA  000074                   R$DTE   000014                   T.LNK   000000                   T3.GFL= 000010
G$$TSS= 000000                   H$X29C  000040                   R$LEN   000024                   T.MXSZ  000050                   T3.MCR= 004000
G$$TTK= 000000                   I$$RAR= 000000                   R$LNK   000000                   T.NAM   000006                   T3.NET= 000020
G$$WRD= 000000                   I$$RDN= 000000                   R$NAM   000002                   T.OFF   000066                   T3.NSD= 000200
HF$DLM= 000002                   KISAR6= ****** GX                R$OWN   000010                   T.PCB   000046                   T3.PMD= 040000
HF$GWY= 000010                   K$$CNT= 177546                   R$$DER= 000000                   T.PRI   000002                   T3.PRV= 010000
HF$HOS= 000004                   K$$CSR= 177546                   R$$K11= 000001                   T.RCVL  000012                   T3.REM= 020000
HF$XDF= 000020                   K$$LDC= 000000                   R$$SND= 000000                   T.RRFL  000072                   T3.ROV= 000040
H$CUG   000010                   K$$TPS= 000074                   R$$11M= 000000                   T.SAST  000054                   T3.RST= 000400
H$DST   000012                   LD$LP = 000000                   S$$WRG= 000000                   T.SRCT  000071                   T3.SLV= 002000
H$D29   000014                   L$$ASG= 000000                   S$$YSZ= 007600                   T.STAT  000032                   T3.SWS= 000002
H$FLG   000000                   L$$DRV= 000000                   TKTCB = ****** GX                T.ST2   000034                   V$$CTR= 001000
H$GLEN  000104                   L$$P11= 000001                   TS.BLK= 171700                   T.ST3   000036                   X$$DBT= 000000
H$GLT   000044                   L$$11R= 000000                   TS.CKP= 000200                   T.TCBL  000030                   $CALLX= ****** GX
H$GNAM  000050                   M$$CRB= 000124                   TS.CKR= 000100                   T.TIO   000057                   $CEACC= ****** GX
H$GNML= 000020                   M$$CRX= 000000                   TS.EXE= 100000                   T.TKSZ  000060                   $XLINK= ****** GX
H$GPT   000046                   M$$FCS= 000000                   TS.HLD= 002000                   T.UCB   000026                   $$$    = 000062
H$HITS  000034                   M$$MGE= 000000                   TS.MSG= 020000                   T2.ABO= 000100                   $$$R   = 000010
H$HLEN  000044                   M$$NET= 000000                   TS.NRP= 010000

. ABS.  000104    000   (RW,I,GBL,ABS,OVR)
        000112    001   (RW,I,LCL,REL,CON)
$$$BYE  000002    002   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 11881 Words  ( 47 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:08.15
SY:PSIBYE.V2,[135,134]PSIBYE/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]PSIBYE
```

```
                                                 G 14
PSIBYE     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
                                                                 F 15
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04  Page 8-1
Symbol table

AF.INI= 000001          MB$ENA  000000          MC$NPR  001630          MD$DMF  000022          ME$EXE  000200
AF.OVR= 000002          MB$FUL  000002          MC$NRE  001200          MD$DMR  000050          ME$LIN  000001
ALCHIN  000000RG        MB$MIX  000002          MC$NRT  001166          MD$DMV  000042          ME$LOG  000002
A$ALI   000012          MB$ONE  000001          MC$NVR  001642          MD$DN   000016          ME$MOD  000004
A$BUF   000006          MB$RXO  000001          MC$P08  000000          MD$DP   000000          ME$NOD  000000
A$DES   000022          MB$TXO  000000          MC$P11  000001          MD$DPV  000044          ME$NON  177777
A$FLG   000000          MB$ZER  000000          MC$RFL  002046          MD$DQ   000006          ME$OBJ  000007
A$LEN   000103          MC$BAB  006073          MC$SBU  002051          MD$DTE  000024          ME$OB2  000004
A$NUM   000010          MC$BID  001765          MC$SFL  002044          MD$DU   000002          ME$PRO  000005
A$OPT   000002          MC$BMC  001767          MC$SYC  000012          MD$DUP  000012          ME$SYS  000006
A$PAR   000004          MC$BSC  001766          MC$SYL  000036          MD$DV   000026          ME.BLO  177744
A$SCD   000020          MC$CAC  001442          MC$SYR  000050          MD$DZ   000030          ME.CON  177753
A$TYP   000011          MC$CAP  001440          MC$SYS  000024          MD$FUL  000000          ME.CST  177765
A$$CHK= 000000          MC$CCL  001445          MC$SYZ  000000          MD$HAL  000001          ME.DIS  177755
A$$CPS= 000000          MC$CDC  002045          MC$TBR  001140          MD$HEL  000002          ME.DON  177600
A$$PRI= 000000          MC$CDP  001441          MC$TBS  001141          MD$KCP  000013          ME.FCO  177762
A$$TRP= 000000          MC$CIF  001465          MC$TDR  002053          MD$KDP  000034          ME.FIO  177756
CF$VR2= ****** GX       MC$CLD  001464          MC$TIM  000000          MD$KDZ  000036          ME.FOP  177763
CHKACC  000470R         MC$COU  100000          MC$TYP  007777          MD$KL   000040          ME.FOR  177776
CHKALI= ****** GX       MC$CTL  001454          MC$UBU  002052          MD$KMX  000054          ME.FUN  177777
CHKDES  000272R         MC$CTR  001452          MC$UFD  002047          MD$KMY  000052          ME.GRD  177745
CHKPAR= ****** GX       MC$CTS  001453          MC$UMR  001132          MD$LEN  000050          ME.HAR  177750
CHKSCO= ****** GX       MC$DOV  002050          MC$UMS  001133          MD$PCL  000011          ME.IID  177767
CURCTX= ****** GX       MC$LBR  001750          MC$VAX  000003          MD$QNA  000005          ME.LCO  177766
C$FLAG= ****** GX       MC$LBS  001751          MC$WID  060000          MD$UNA  000001          ME.LPR  177757
C$$CKP= 000000          MC$LDI  001774          MC$WIH  040000          MD$WIT  000002          ME.MPR  177773
C$$DRE= 000400          MC$LDO  001775          MC$WIL  020000          MD.ABD  000017          ME.MVE  177771
C$$RSH= 177564          MC$LDR  001762          MC$W08  020000          MD.ACC  000010          ME.OPE  177747
D$$BUG= 177514          MC$LDS  001763          MC$W16  040000          MD.AOB  000016          ME.PLD  177751
D$$ISK= 000000          MC$LLB  002021          MC$W32  060000          MD.BOB  000011          ME.PMI  177743
D$$L11= 000001          MC$LLP  002115          MC$XBR  001750          MD.DIA  000007          ME.PNA  177752
D$$YNC= 000000          MC$LLR  002007          MC$XBS  001751          MD.DOB  000015          ME.PRI  177775
D$$YNM= 000000          MC$LOO  000001          MC$XCJ  000322          MD.DUM  000002          ME.PTY  177772
E$$XPR= 000000          MC$LRB  002020          MC$XCR  002260          MD.FAI  000014          ME.PVA  177760
F$$LVL= 000001          MC$LRP  002114          MC$XCS  002261          MD.FNA  000001          ME.RES  177761
G$$TPP= 000000          MC$LRR  002006          MC$XDR  001762          MD.FOB  000006          ME.ROD  177754
G$$TSS= 000000          MC$LSI  002032          MC$XDS  001763          MD.LOA  000001          ME.SIZ  177774
G$$TTK= 000000          MC$LST  002033          MC$XFR  002272          MD.LSH  000020          ME.SYS  177746
G$$WRD= 000000          MC$LUP  006072          MC$XFS  002273          MD.NOB  000012          ME.UCO  177770
I$$RAR= 000000          MC$MAP  010000          MC$XLJ  000323          MD.NDN  177777          MF$ACT  177776
I$$RDN= 000000          MC$MBL  001764          MC$XLR  002330          MD.PER  000000          MF$ADD  000000
K$$CNT= 177546          MC$MBX  004420          MC$XMA  000310          MD.RES  000004          MF$ADJ  177774
K$$CSR= 177546          MC$MBY  001752          MC$XMC  002305          MD.ROB  000005          MF$ALL  177775
K$$LDC= 000000          MC$MPX  004406          MC$XMS  002304          MD.RSH  000013          MF$BYE  000300
K$$TPS= 000074          MC$NAP  001604          MC$XNR  002332          MD.SDU  000005          MF$CHA  000023
LC$EXT  000000          MC$NBR  001130          MC$XRC  002316          MD.SLO  000003          MF$DUM  000020
LC$INT  000001          MC$NBS  001131          MC$XRR  002331          MD.SNA  000000          MF$EVT  000001
LD$LP = 000000          MC$NCR  001154          MC$XRS  002342          MD.TLO  000004          MF$FUP  000264
L$$ASG= 000000          MC$NCS  001155          MC$020  000002          MD.UNA  000002          MF$KNO  177777
L$$DRV= 000000          MC$NML  001274          MD$CI   000007          MD.UNR  000003          MF$LOA  000017
L$$P11= 000001          MC$NMR  001142          MD$CNA  000003          MD.UOB  000007          MF$LOO  177775
L$$11R= 000000          MC$NMS  001143          MD$COU  000001          MD.VOL  000006          MF$REA  000024
MAPADD= ****** GX       MC$NNO  001606          MD$DA   000010          ME$ALI  000010          MF$SIG  177773
MAXACC= 000020          MC$NNU  001605          MD$DL   000004          ME$AL2  000003          MF$SPF  000302
MAXID = 000020          MC$NOP  001607          MD$DLV  000020          ME$ARE  000005          MF$SYS  000026
MAXPAS= 000010          MC$NDR  000000          MD$DMC  000014          ME$CIR  000003          MF$TES  000022
MB$DIS  000001          MC$NPF  001616          MD$DMF  000046          ME$EXA  000000          MF$TRA  000301
```

```
                                                                 G 15
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04  Page 8-2
Symbol table
```

```
                                             .SBTTL  CLRALI - CLEAR AN ALIAS
                                     ;+
                                     ; **-CLRALI-CLEAR AN ALIAS BLOCK
                                     ;
                                     ; THIS ROUTINE IS CALLED TO CLEAR ONE OR MORE ALIAS NAME BLOCKS (ANB'S).
                                     ;
                                     ; INPUTS:
                                     ;        R4 = ADDRESS OF CONTEXT AREA WITH THE FOLLOWIN OFFSETS FILLED IN:
                                     ;               A$TYP = ALIAS NAME FORMAT TYPE:
                                     ;                          -3 = ALL (MF$ALL)
                                     ;                          -2 = ACTIVE (MF$ACT)
                                     ;                          -1 = KNOWN (MF$KNO)
                                     ;                          >0 = SPECIFIC ALIAS
                                     ;               A$ALI = ALIAS NAME, SPACE FILLED IF A$TYP > 0
                                     ;               A$SCO = SCOPE OF ALIAS(ES)
                                     ;
                                     ; OUTPUTS:
                                     ;        R0 = RETURN CODE
                                     ;
                                     ; REGISTERS:
                                     ;        R4 IS PRESERVED
                                     ;
                                     ;-
000336                               CLRALI:
000336  012700  000001                       MOV     #MS.SUC,R0              ; ASSUME SUCCESS

                                             .IF DF  R$$PRO                  ; PRO/DECNET

                                             MOV     #DF.AUP,R1              ; SET ALIAS UPDATE BIT
                                             CALL    CKLOCK                  ; WAIT UNTIL NT.CFG PARTITION IS UNLOCKED
                                             BCC     10$                     ; IF CC, UNLOCKED
                                             MOV     #ME.OPE,R0              ; ELSE OPERATION FAILURE
                                             RETURN

                                             .ENDC   ; IF DF R$$PRO

000342                               10$:    SWSTK$  60$                     ;; ENTER SYSTEM STATE
000346                                       SAVMAP                          ;; SAVE CURRENT MAPPING
000352  017700  000000G                      MOV     @DECPT,R0               ;; POINT TO HOME BLOCK
000356  001463                               BEQ     50$                     ;; NO HOME BLOCK, NO ALIASES
000360  062700  000000                       ADD     #D$ANN,R0               ;; POINT TO ALIAS NAME LISTHEAD
000364  105764  000011                       TSTB    A$TYP(R4)               ;; CLEAR A SPECIFIED ALIAS ?
000370  002426                               BLT     25$                     ;; IF LT, NO
                                     ;
                                     ; CLEAR A SPECIFIED ALIAS
                                     ;
000372                                       CALL    FINDA                   ;; LOOK FOR ALIAS NAME BLOCK
000376  103020                               BCC     20$                     ;; IF CC, FOUND IT - DELETE IT
000400  012766  177770  000004               MOV     #ME.UCO,2+BIAS(SP)      ;; ELSE, SET ERROR CODE IN SAVED R0
000406  012764  000010  000004               MOV     #ME$ALI,A$PAR(R4)       ;; SET ERROR DETAIL
000414  016703  000000G                      MOV     CURCTX,R3               ;; GET ADDRESS OF CONTEXT AREA
000420  032763  000000G 000000G              BIT     #CF$VR2,C$FLAG(R3)      ;; CONNECTED TO V2.0 NCP?
000426  001437                               BEQ     50$                     ;; BR IF YES
000430  012764  000003  000004               MOV     #ME$AL2,A$PAR(R4)       ;; ELSE SET V2.0 ENTITY IO
000436  000433                               BR      50$                     ;; AND RETURN
000440                               20$:    CALL    DEAANB                  ;; UNLINK AND DEALLOC ALIAS NAME BLOCK
000444  000430                               BR      50$                     ;; AND RETURN
```

```
    677 001032  012713  101000                 MOV     #CS.ERR+CS.FTL,(R3)     ;; Tell driver a fatal error occurred
    678 001036  012743  000026                 MOV     #FC.MLD+FS.HLT,-(R3)    ;; (Sub)function = load halted
    679 001042  010446                         MOV     R4,-(SP)                ;; Save CCB address
    680 001044                                 CALL    @PDSPL                  ;; Dispatch to driver w/priority
    681 001050  012604                         MOV     (SP)+,R4                ;; Recover CCB address
    682 001052                                 CALL    @CCBRT                  ;; Return CCB to pool
    683 001056  000750                         BR      10$                     ;; Dequeue another load request
    684
    685 001060                         30$:    RETURN                          ;; Return to user state at "40$"
    686 001062  005067  000156'        40$:    CLR     CURQST                  ;  No more requests outstanding
    687 001066                                 EXIT$S                          ;  Program exit
    688
    689                                        .IFT                            ; If system is RSX-11M  ...
```

```
                                                                                G 2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

DSPDRV    000414 R          9-466     #10-499     10-560     18-980
EF$ACT  = 000001          #6-83
ETYPCL  = 010500          #7-143      11-617
EVTBLK    000306 R        #8-216      11-606      11-615     11-617
EVTEFL  = 000002          #7-101      11-617
EVTLUN  = 000002          #7-100      9-269       11-617
EV$ACF  = 000201          #6-83
EV$ADR  = 000420          #6-83
EV$ADU  = 000417          #6-83
EV$APL  = 000400          #6-83
EV$ARC  = 000421          #6-83
EV$AUC  = 000010          #6-83
EV$AUS  = 000003          #6-83
EV$CDF  = 000520          #6-83
EV$COZ  = 000011          #6-83
EV$DBR  = 000302          #6-83
EV$GAS  = 035101          #6-83
EV$HCE  = 035114          #6-83
EV$HCI  = 035113          #6-83
EV$HFE  = 000506          #6-83
EV$IFL  = 000413          #6-83
EV$IFO  = 000415          #6-83
EV$IFS  = 000414          #6-83
EV$INF  = 000515          #6-83
EV$LDL  = 000407          #6-83
EV$LDN  = 010416          #6-83
EV$LDO  = 000411          #6-83
EV$LDS  = 000410          #6-83
EV$LSC  = 000500          #6-83
EV$LUP  = 000412          #6-83
EV$NOL  = 000402          #6-83
EV$NRC  = 000416          #6-83
EV$NSC  = 000200          #6-83
EV$NUL  = 000401          #6-83
EV$NVR  = 000406          #6-83
EV$OPL  = 000403          #6-83
EV$PCC  = 034000          #6-83
EV$PCI  = 034001          #6-83
EV$PCM  = 034002          #6-83
EV$PFE  = 000404          #6-83
EV$PPC  = 034003          #6-83
EV$RCF  = 000517          #6-83
EV$RDC  = 010001          #6-83
EV$RDR  = 010002          #6-83
EV$RJE  = 035106          #6-83
EV$RSC  = 000501          #6-83
EV$RUL  = 000405          #6-83
EV$SNA  = 035000          #6-83
EV$SNF  = 000516          #6-83
EV$SPE  = 035001          #6-83
EV$XCE  = 034110          #6-83
EV$XDI  = 013600          #6-83
```

```
                                                                                H 2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31     PAGE 3
SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
000042  000000G         STPCT:: .WORD   $STPCT
000044  000000G         TKTCB:: .WORD   $TKTCB
000046  000000G         CCBAF:: .WORD   $CCBAF
000050  000000G         CCBAL:: .WORD   $CCBAL
000052  000000G         CCBCT:: .WORD   $CCBCT
000054  000000G         CCBGT:: .WORD   $CCBGT
000056  000000G         CCBRT:: .WORD   $CCBRT
000060  000000G         CCBNM:: .WORD   $CCBNM
000062  000000G         CEACC:: .WORD   $CEACC
000064  000000G         CSBRT:: .WORD   $CSBRT
000066  000000G         DDMAN:: .WORD   $DDMAN
000070  000000G         DECPT:: .WORD   $DECPT
000072  000000G         FILHD:: .WORD   $FILHD
000074  000000G         LDBAF:: .WORD   $LDBAF
000076  000000G         LDBGT:: .WORD   $LDBGT
000100  000000G         LDBRT:: .WORD   $LDBRT
000102  000000G         LGCON:: .WORD   $LGCON
000104  000000G         LGFNB:: .WORD   $LGFNB
000106  000000G         LGMON:: .WORD   $LGMON
000110  000000G         LGSTT:: .WORD   $LGSTT
000112  000000G         LGUIC:: .WORD   $LGUIC
000114  000000G         LLCRS:: .WORD   $LLCRS
000116  000000G         LLCTA:: .WORD   $LLCTA
000120  000000G         NMCLH:: .WORD   $NMCLH

                        .IF DF R$$PRO   ; PRO/DECnet

                        NMCL2:: .WORD   $NMCLH+2

                        .IFF    ; DF R$$PRO

000122  000000G         NMCL2:: .WORD   $NMCL2

                        .ENDC   ; DF R$$PRO

000124  000000G         NMCRS:: .WORD   $NMCRS
000126  000000G         NMLST:: .WORD   $NMLST
000130  000000G         NTLPT:: .WORD   $NTLPT
000132  000000G         OBJHD:: .WORD   $OBJHD
000134  000000G         PDVID:: .WORD   $PDVID
000136  000000G         PDVNM:: .WORD   $PDVNM
000140  000000G         PDVTA:: .WORD   $PDVTA
000142  000000G         PSIPT:: .WORD   $PSIPT
000144  000000G         RDBAF:: .WORD   $RDBAF
000146  000000G         RDBCT:: .WORD   $RDBCT
000150  000000G         RDBNM:: .WORD   $RDBNM
000152  000000G         RDBSZ:: .WORD   $RDBSZ
000154  000000G         RDBTH:: .WORD   $RDBTH
000156  000000G         SDBAF:: .WORD   $SDBAF
000160  000000G         SDBCT:: .WORD   $SDBCT
000162  000000G         SDBNM:: .WORD   $SDBNM
000164  000000G         SLTMA:: .WORD   $SLTMA
000166  000000G         SLTNM:: .WORD   $SLTNM
000170  000000G         SNAPT:: .WORD   $SNAPT
000172  000000G         XAVL::  .WORD   $XAVL
000174  000000G         XBIAS:: .WORD   $XBIAS
000176  000000G         ZTIME:: .WORD   $ZTIME
```

```
                                        .SBTTL  FINREQ - FINISH REQUEST
                                ;+
                                ; **-FINREQ-FINISH REQUEST
                                ;
                                ; THIS ROUTINE IS CALLED TO SET UP RETURN CODES.
                                ;
                                ; INPUTS:
                                ;       R0 = NICE RETURN CODE
                                ;       R2 = LENGTH OF DATA BEING RETURNED
                                ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
                                ;
                                ; OUTPUTS:
                                ;       R0 = RETURN CODE (NICE CODE IN HIGH BYTE)
                                ;       R1 = LENGTH OF DATA BEING RETURNED
                                ;       R3 = I/O PACKET ADDRESS
                                ;-

002736                          FINREQ:
002736  012746  000001                  MOV     #IS.SUC,-(SP)           ; INIT STATUS WORD

                                        .IF DF  R$$PRO                  ; PRO/DECNET

                                        TST     R0                      ; WAS NICE CODE SUCCESSFUL ?
                                        BLE     10$                     ; IF LE, NO
                                        MOV     @DECPT,R4               ; GET DECNET HOME BLOCK ADDRESS ; RJK02
                                        BIT     #DF.CFN,D$STAT(R4)      ; CHECK CONFIGURATION FILE UPDATE
                                        BEQ     5$                      ; IF EQ, NOT SET
                                        BIS     #100,R0                 ; ELSE SET BIT IN NICE RETURN CODE BYTE
                                5$:     BIT     #DF.AFN,D$STAT(R4)      ; CHECK ALIAS FILE UPDATE
                                        BEQ     10$                     ; IF EQ, NOT SET
                                        BIS     #40,R0                  ; ELSE SET BIT IN NICE RETURN CODE BYTE

                                        .ENDC

002742  110066  000001          10$:    MOVB    R0,1(SP)                ; PUT RETURN STATUS IN HIGH BYTE
002746  012600                          MOV     (SP)+,R0                ; GET RETURN STATUS
002750  010201                          MOV     R2,R1                   ; SET LENGTH OF BUFFER RETURNED
002752  016703  175232                  MOV     CURIOP,R3               ; RESTORE I/O PACKET ADDRESS
002756                                  RETURN

        001472'                         .END    MANACP
```

```
                                                                          G 5
**FILE**ID**MANAST









MM      MM     AAAAAA     NN      NN     AAAAAA       SSSSSSSS   TTTTTTTTTT
MM      MM     AAAAAA     NN      NN     AAAAAA       SSSSSSSS   TTTTTTTTTT
MMMM  MMMM   AA      AA   NN      NN   AA      AA   SS               TT
MMMM  MMMM   AA      AA   NN      NN   AA      AA   SS               TT
MM  MM  MM   AA      AA   NNNN    NN   AA      AA   SS               TT
MM  MM  MM   AA      AA   NNNN    NN   AA      AA   SS               TT
MM      MM   AA      AA   NN  NN  NN   AA      AA     SSSSSS         TT
MM      MM   AA      AA   NN  NN  NN   AA      AA     SSSSSS         TT
MM      MM   AAAAAAAAAA   NN    NNNN   AAAAAAAAAA           SS       TT
MM      MM   AAAAAAAAAA   NN    NNNN   AAAAAAAAAA           SS       TT
MM      MM   AA      AA   NN      NN   AA      AA           SS       TT          ....
MM      MM   AA      AA   NN      NN   AA      AA           SS       TT          ....
MM      MM   AA      AA   NN      NN   AA      AA   SSSSSSSS         TT          ....
MM      MM   AA      AA   NN      NN   AA      AA   SSSSSSSS         TT          ....








LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                          H 5
MANAST - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:29
Table of contents
```

```
     44                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     45                                   ;
     46                                   ; MACRO LIBRARY CALLS
     47                                   ;
     48                                          .MCALL  GTSK$S,EXTK$S,CALLR,SAVRG,RESRG,PKTDF$,SREA$S,CCBDF$
     49
     50 000000                                   PKTDF$                          ; DEFINE I/O PACKET OFFSETS
     51 000000                                   CCBDF$                          ; DEFINE COMMUNICATIONS CONTROL BLOCK SYMBOLS
     52
     53                                   ;
     54                                   ; LOCAL DEFINITIONS
     55                                   ;
     56          000000                          MN$VOL=0                        ; ASSEMBLE FOR VOLATILE ACP
     57
```

```
MANBUF     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              6-84        6-100       6-120       7-230       7-232       8-264       8-279       9-306       9-313       10-353
                  10-358      10-362      11-395      11-398
CALLR            #5-48        10-371      11-404
CCBDF$           #5-48        5-51
DIR$             #6-127       6-127      #6-134       6-134      #7-212       7-212
EXTK$S           #5-48        7-212
GTSK$S           #5-48        6-127
MOV$             #6-127       6-127      #6-134       6-134      #7-212       7-212
OFF$             #6-127
PKTDF$           #5-48        5-50
RESRG            #5-48
RETURN            6-124       6-136       7-242       8-280       9-323
SAVRG            #5-48
SOB               7-229       9-322
SREA$S           #5-48        6-134
SWSTK$            6-84        7-230
```

```
**FILE**ID**MANBYE
```

```
                                                                  G  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-3
Symbol table

MP$DWE  001321          MP$MLN  001631          MP$PRO  002130          MP$TST  000144          MT$ASC  000100
MP$ELT  000157          MP$MLP  006333          MP$PSS  001763          MP$TYP  002130          MT$BIL  000001
MP$ETY  001605          MP$MRB  002171          MP$PST  001762          MP$UCS  004407          MT$BYS  000011
MP$EVE  000311          MP$MRO  001605          MP$RET  001631          MP$USR  001750          MT$CI   000007
MP$FNC  001752          MP$MRP  006322          MP$RFA  001323          MP$VEC  004410          MT$CLE  000077
MP$GDT  002222          MP$MRT  002153          MP$RMX  002201          MP$VER  004406          MT$COD  000200
MP$GNM  002223          MP$MRV  006320          MP$RPA  004374          MP$WDF  002165          MT$CON  000001
MP$GRO  002115          MP$MSB  000170          MP$RPR  001606          MP$WMX  002177          MT$DMC  000004
MP$GRP  000541          MP$MVE  000145          MP$RRT  002213          MP$XMT  002166          MT$ETH  000006
MP$GTY  002224          MP$MVI  001634          MP$RST  002212          MP$XPF  000242          MT$HEX  000040
MP$HAD  001757          MP$MVR  001751          MP$RSV  001754          MP$XPL  000244          MT$LPB  000005
MP$HBT  002142          MP$MWN  002154          MP$RSZ  001756          MP$XPT  000240          MT$MAX  000037
MP$HDD  000162          MP$MXB  002172          MP$RTI  001616          MP$XXX  177777          MT$MUL  000100
MP$HTM  001612          MP$MXC  000466          MP$RTM  001611          MP$5ST  004406          MT$NLE  000017
MP$HWA  002210          MP$MXR  001630          MP$RVE  001604          MSVRUN= 000014          MT$NON  000001
MP$IAT  001322          MP$MXW  002143          MP$RVT  000156          MSVSUB= 005240          MT$NR4  000005
MP$IDE  000144          MP$NAA  006335          MP$SAC  000514          MSVUNS= 000015          MT$NTY  000060
MP$IDP  006327          MP$NAC  001130          MP$SAD  000543          MS$ACT  000001          MT$OCT  000060
MP$IHO  000215          MP$NAP  006326          MP$SCA  000310          MS$ADU  000010          MT$PHA  000002
MP$INA  002177          MP$NCT  000240          MP$SCO  000144          MS$ALO  000007          MT$POI  000000
MP$IND  002204          MP$NET  002114          MP$SCT  002176          MS$ASE  000006          MT$QP2  000010
MP$INI  002201          MP$NLI  000765          MP$SDU  000203          MS$ATR  000011          MT$ROU  000000
MP$ITI  000776          MP$NNA  000764          MP$SDV  000160          MS$AUT  000000          MT$RO4  000004
MP$LAA  000231          MP$NOD  000500          MP$SEH  004401          MS$CLE  000003          MT$SEC  000000
MP$LAN  000234          MP$NRA  006334          MP$SER  000144          MS$DED  000004          MT$SGD  000020
MP$LAR  000202          MP$NRB  000214          MP$SET  000000          MS$DIE  000003          MT$SYS  000002
MP$LAS  000012          MP$NSA  001617          MP$SGZ  001644          MS$DUM  000004          MT$TER  000001
MP$LBS  000203          MP$NTI  002141          MP$SID  000176          MS$FAI  000013          MT$TRI  000002
MP$LCO  000226          MP$NUM  001642          MP$SIN  000310          MS$GLO  000000          MT$TYP  007777
MP$LCT  000156          MP$NVE  001274          MP$SLI  000156          MS$HOL  000002          MT$USD  000000
MP$LHL  000232          MP$NXN= 001476          MP$SLO  000171          MS$INA  000002          MT$X25  000003
MP$LLE  000227          MP$OAC  000632          MP$SMX  002202          MS$LOA  000003          MU$INC  000001
MP$LLO  004432          MP$OAN  000620          MP$SND  000500          MS$LOO  000002          MU$OUT  000002
MP$LMB  002152          MP$OCO  000776          MP$SNP  006332          MS$OFF  000001          MU$PER  000000
MP$LMX  002200          MP$OHO  000214          MP$SNU  000542          MS$ON   000000          MV$III  000000
MP$LNA  000144          MP$ONA  000764          MP$SOB  000524          MS$PRO  000006          MV$IV   000001
MP$LNO  000233          MP$ONR  002114          MP$SPA  000157          MS$REA  000004          MXACC = 000020
MP$LOA  000170          MP$OQL  001453          MP$SPR  000536          MS$REF  000001          MXDSTN= 000020
MP$LOG  004514          MP$OTI  000777          MP$SPS  000513          MS$RES  000003          MXGRPN= 000020
MP$LOO  000620          MP$OTY  001022          MP$STA  000000          MS$RST  000001          MXID  = 000020
MP$LTM  001613          MP$OUS  000777          MP$STI  002140          MS$RSX  000002          MXOBJN= 000020
MP$LTY  004533          MP$OVE  001010          MP$STT  002201          MS$RT   000005          MXPAS = 000010
MP$LWI  000230          MP$OWN  004374          MP$STY  000175          MS$SER  000002          MXUID = 000020
MP$MAC  004420          MP$PAR  000036          MP$SUB  000001          MS$SHU  000002          MX$ACT  000020
MP$MAD  001630          MP$PAS  000513          MP$SUR  000156          MS$SRV  000007          MX$CIR  000020
MP$MAP  006323          MP$PCH  002152          MP$SUS  000512          MS$STA  000000          MX$CLN  000020
MP$MAR  001635          MP$PCO  000024          MP$SVR  000163          MS$SYN  000012          MX$CNM  000006
MP$MAV  006321          MP$PCT  000144          MP$SWI  002260          MS$TER  000001          MX$CON  000006
MP$MBN  001636          MP$PDT  002114          MP$SYP  002424          MS$TOP  000003          MX$DAC  000020
MP$MBR  001637          MP$PHA  000170          MP$TFL  000156          MS$TRI  000005          MX$DTE  000020
MP$MBU  001642          MP$PHY  000012          MP$TH1  002206          MS$UNR  000005          MX$FIL  000034
MP$MCB  000156          MP$PLI  000025          MP$TH2  002205          MS$VMS  000004          MX$LGC  000777
MP$MCO  001632          MP$PLN  002140          MP$TH3  002202          MS.DON  177600          MX$LIN  000020
MP$MDE  004421          MP$PLO  000012          MP$TLN  000202          MS.MOR  000002          MX$LON  000006
MP$MHO  001633          MP$PMC  004374          MP$TLO  000172          MS.PAR  000003          MX$NMS  000454
MP$MLB  000202          MP$PNT  002126          MP$TPA  004375          MS.SUC  000001          MX$NOD  000006
MP$MLK  001306          MP$PRI  004411          MP$TRI  002164          MT$AR4  000003          MX$OBJ  000006
```

```
                                                                  H  8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-4
Symbol table
```

```
MANFRK      CREATED BY  MACRO  ON 29-JUN-85 AT 10:30      PAGE 2

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME       REFERENCES

CALL              6-101       6-141       7-189       7-207       7-213
CALLR             6-148       6-149       7-216
PKTDF$           #5-54        5-57
RETURN            7-179       7-217
SWSTK$            6-141       7-213
TCBDF$           #5-54        5-56
```

```
**FILE**ID**MANQIO
```

```
Symbol table

ALCHIN   000420RG       LC$INT   000001         MC$LUP   006072         MC$XFS   002273         MD.NOB   000012
ALCHOP   000722RG       LD$LP =  000000         MC$MAP   010000         MC$XLJ   000323         MD.NON   177777
ALID     000520R        LICHIN   000464RG       MC$MBL   001764         MC$XLR   002330         MD.PER   000000
ALREIN   000426RG       LICHOP   000722RG       MC$MBX   004420         MC$XMA   000310         MD.RES   000004
ALREOP   000722RG       LIID     000520R        MC$MBY   001752         MC$XMC   002305         MD.ROB   000005
ALUN     000000R        LIREIN   000472RG       MC$MPX   004406         MC$XMS   002304         MD.RSH   000013
ARID     000652R        LIREOP   000722RG       MC$NAP   001604         MC$XNR   002332         MD.SDU   000005
ARREIN   000434RG       LIZEIN   000500RG       MC$NBR   001130         MC$XRC   002316         MD.SLO   000003
ARREOP   000722RG       LIZEOP   000722RG       MC$NBS   001131         MC$XRR   002331         MD.SNA   000000
A$$CHK=  000000         LOCHIN   000532RG       MC$NCR   001154         MC$XRS   002342         MD.TLO   000004
A$$CPS=  000000         LOCHOP   000722RG       MC$NCS   001155         MC$020   000002         MD.UNA   000002
A$$PRI=  000000         LOID     000544R        MC$NML   001274         MD$CI    000007         MD.UNR   000003
A$$TRP=  000000         LOREIN   000540RG       MC$NMR   001142         MD$CNA   000003         MD.UOB   000007
A.LULU=  000002         LOREOP   000722RG       MC$NMS   001143         MD$COU   000001         MD.VOL   000006
A.LUNA=  000004         L$$ASG=  000000         MC$NNO   001606         MD$DA    000010         ME$ALI   000010
A.LUNU=  000006         L$$DRV=  000000         MC$NNU   001605         MD$DL    000004         ME$AL2   000003
CICHIN   000442RG       L$$P11=  000001         MC$NDP   001607         MD$DLV   000020         ME$ARE   000005
CICHOP   000722RG       L$$11R=  000000         MC$NOR   000000         MD$DMC   000014         ME$CIR   000003
CIID     000520R        MB$DIS   000001         MC$NPF   001616         MD$DMF   000046         ME$EXA   000000
CIREIN   000450RG       MB$ENA   000000         MC$NPR   001630         MD$DMP   000022         ME$EXE   000200
CIREOP   000722RG       MB$FUL   000002         MC$NRE   001200         MD$DMR   000050         ME$LIN   000001
CIZEIN   000456RG       MB$MIX   000002         MC$NRT   001166         MD$DMV   000042         ME$LOG   000002
CIZEOP   000722RG       MB$ONE   000001         MC$NVR   001642         MD$DN    000016         ME$MOD   000004
CM$RSX=  000200         MB$RXO   000001         MC$P08   000000         MD$DP    000000         ME$NOD   000000
C$$CKP=  000000         MB$TXO   000000         MC$P11   000001         MD$DPV   000044         ME$NON   177777
C$$ORE=  000400         MB$ZER   000000         MC$RFL   002046         MD$DQ    000006         ME$OBJ   000007
C$$RSH=  177564         MC$BAB   006073         MC$SBU   002051         MD$DTE   000024         ME$OB2   000004
D$$BUG=  177514         MC$BID   001765         MC$SFL   002044         MD$DU    000002         ME$PRO   000005
D$$ISK=  000000         MC$BMC   001767         MC$SYC   000012         MD$DUP   000012         ME$SYS   000006
D$$L11=  000001         MC$BSC   001766         MC$SYL   000036         MD$DV    000026         ME.BLO   177744
D$$YNC=  000000         MC$CAC   001442         MC$SYR   000050         MD$DZ    000030         ME.CON   177753
D$$YNM=  000000         MC$CAP   001440         MC$SYS   000024         MD$FUL   000000         ME.CST   177765
ERRTBL   000014R        MC$CCL   001445         MC$SYZ   000000         MD$HAL   000001         ME.DIS   177755
E$$XPR=  000000         MC$CDC   002045         MC$TBR   001140         MD$HEL   000002         ME.DON   177600
F$$LVL=  000001         MC$CDP   001441         MC$TBS   001141         MD$KCP   000013         ME.FCO   177762
G$$TPP=  000000         MC$CIF   001465         MC$TDR   002053         MD$KDP   000034         ME.FIO   177756
G$$TSS=  000000         MC$CLD   001464         MC$TIM   000000         MD$KDZ   000036         ME.FOP   177763
G$$TTK=  000000         MC$COU   100000         MC$TYP   007777         MD$KL    000040         ME.FOR   177776
G$$WRD=  000000         MC$CTL   001454         MC$UBU   002052         MD$KMX   000054         ME.FUN   177777
IE.ABO=  ****** GX      MC$CTR   001452         MC$UFD   002047         MD$KMY   000052         ME.GRO   177745
IE.DAO=  ****** GX      MC$CTS   001453         MC$UMR   001132         MD$LEN   000050         ME.HAR   177750
IE.HWR=  ****** GX      MC$DOV   002050         MC$UMS   001133         MD$PCL   000011         ME.IID   177767
IE.IDU=  ****** GX      MC$LBR   001750         MC$VAX   000003         MD$QNA   000005         ME.LCO   177766
IE.NDR=  ****** GX      MC$LBS   001751         MC$WID   060000         MD$UNA   000001         ME.LPR   177757
IE.NFW=  ****** GX      MC$LDI   001774         MC$WIH   040000         MD$WIT   000002         ME.MPR   177773
IE.NLN=  177733         MC$LDO   001775         MC$WIL   020000         MD.ABO   000017         ME.MVE   177771
IE.NST=  ****** GX      MC$LDR   001762         MC$W08   020000         MD.ACC   000010         ME.OPE   177747
IE.OFL=  ****** GX      MC$LDS   001763         MC$W16   040000         MD.ADB   000016         ME.PLO   177751
IE.PRI=  ****** GX      MC$LLB   002021         MC$W32   060000         MD.BDB   000011         ME.PMI   177743
INIQIO   000070R        MC$LLP   002115         MC$XBR   001750         MD.DIA   000007         ME.PNA   177752
I$$RAR=  000000         MC$LLR   002007         MC$XBS   001751         MD.DOB   000015         ME.PRI   177775
I$$RDN=  000000         MC$LOO   000001         MC$XCJ   000322         MD.DUM   000002         ME.PTY   177772
K$$CNT=  177546         MC$LRB   002020         MC$XCR   002260         MD.FAI   000014         ME.PVA   177760
K$$CSR=  177546         MC$LRP   002114         MC$XCS   002261         MD.FNA   000001         ME.RES   177761
K$$LDC=  000000         MC$LRR   002006         MC$XDR   001762         MD.FOB   000006         ME.ROO   177754
K$$TPS=  000074         MC$LSI   002032         MC$XDS   001763         MD.LOA   000001         ME.SIZ   177774
LC$EXT   000000         MC$LST   002033         MC$XFR   002272         MD.LSH   000020         ME.SYS   177746
```

```
Symbol table
```

```
                                                .SBTTL  TRANAM - TRANSLATE A NAME TO AN ADDRESS
                                        ;+
                                        ; **-TRANAM-TRANSLATE A NAME TO AN ADDRESS
                                        ;
                                        ; THIS ROUTINE TRANSLATES A NAME TO AN ADDRESS
                                        ;
                                        ; INPUTS:
                                        ;       R1 = POINTER TO NAME
                                        ;       R3 = ADDRESS OF BUFFER
                                        ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = RETURN STATUS
                                        ;       R2 = LENGTH OF DATA TO RETURN (0 OR 2)
                                        ;-

000056                                  TRANAM::
000056  005002                                  CLR     R2                      ; ASSUME NO ADDRESS TO RETURN
000060  016705  000000G                         MOV     CURIOP,R5               ; GET ADDRESS OF I/O PACKET
000064  026527  000006G 000006                  CMP     I.PRM+6(R5),#6          ; WAS A SIX BYTE NAME SUPPLIED ?
000072  001403                                  BEQ     10$                     ; IF EQ, YES - TRY TO MAP IT
000074  012700  177776                          MOV     #ME.FOR,R0              ; ELSE, MESSAGE FORMAT ERROR
000100  000414                                  BR      30$                     ; AND RETURN
000102  010100                          10$:    MOV     R1,R0                   ; POINT TO THE NAME
000104                                          CALL    MAPREM                  ; MAP THE NAME TO AN ADDRESS
000110  103003                                  BCC     20$                     ; IF CC, SUCCESS
000112  012700  177770                          MOV     #ME.UCO,R0              ; ELSE "UNKNOWN COMPONENT"
000116  000405                                  BR      30$                     ; ...
000120  012700  000001                  20$:    MOV     #MS.SUC,R0              ; SUCCESS !
000124  010113                                  MOV     R1,(R3)                 ; SET THE ADDRESS IN THE BUFFER
000126  012702  000002                          MOV     #2,R2                   ; TWO BYTES TO RETURN
000132                                  30$:    RETURN
```

```
                                          .SBTTL  INIACP - INITIALIZE ACP
                                   ;+
                                   ; **-INIACP-INITIALIZE ACP
                                   ;
                                   ; THIS ROUTINE IS CALLED TO INITIALIZE THE STATE OF THE ACP. IT
                                   ; CALLS "INIBUF" TO INITIALIZE THE ACP BUFFER AREA.
                                   ;
                                   ; INPUTS:
                                   ;         NONE
                                   ;
                                   ; OUTPUTS:
                                   ;         BUFFER DESCRIPTORS ARE INITIALIZED
                                   ;
                                   ;-

000000                             INIACP::
                                          .IF     DF      R$$MPL
                                            .IF   NDF     R$$PRO
                                                  DIR$    #GETVEC         ; TRANSLATE VECTORS (IF NECESSARY)
                                            .ENDC
                                          .ENDC

000000  005767  000000G                   TST     POOLPT                  ; HAS THE POOL BEEN INITIALIZED ?
000004  001002                            BNE     10$                     ; IF NE, YES
000006                                    CALL    INIBUF                  ; INITIALIZE THE BUFFERS
000012                             10$:   RETURN
```

```
                                   .ENDC
                         ;
                         ; END-OF-TABLE
                         ;
      000460             MNFUN$   FLAGS=RF$END                              ; END OF TABLE

              000001               .END
```

```
                                                                     G 14
PSIBYE      CREATED BY  MACRO  ON 29-JUN-85 AT 10:32       PAGE 1
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE                REFERENCES

DEACB   = ******  GX          6-97
H$RDTE    000004              6-82
KISAR6  = ******  GX          6-78       *6-100
N$$VCT  = ******              6-78        6-89         6-95         6-100
PSIBYE    000000 RG           5-59       #6-76
PSIPT   = ******  GX          6-80
R$LEN     000024              6-96
R$OWN     000010              6-90
TKTCB   = ******  GX          6-84
T.UCB     000026              6-90
$CALLX  = ******  GX          6-95
$CEACC  = ******  GX          6-89
$XLINK  = ******  GX          6-95
```

```
                                                                     H 14
PSIBYE      CREATED BY  MACRO  ON 29-JUN-85 AT 10:32       PAGE 2
MACRO CROSS REFERENCE                                      CREF   04.00
```

```
                                                      G 15
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04  Page 8-2
Symbol table

MF$TRI   000021        MP$AMH   001641        MP$DRO   001441        MP$MBN   001636        MP$PDT   002114
MF$ZER   000025        MP$ANB   004420        MP$DST   000454        MP$MBR   001637        MP$PHA   000170
ML$ALL   100000        MP$ASB   000024        MP$DTE   002140        MP$MBU   001642        MP$PHY   000012
ML$CLS   000000        MP$ASC   004432        MP$DTY=  001452        MP$MCB   000156        MP$PLI   000025
ML$CON   000001        MP$AUS   000512        MP$DUA   000207        MP$MCO   001632        MP$PLN   002140
ML$EXT   000001        MP$BBT   002165        MP$DUC   000210        MP$MDE   004421        MP$PLO   000012
ML$FIL   000002        MP$BDF   002164        MP$DUM   000202        MP$MHO   001633        MP$PMC   004374
ML$FIR   000000        MP$BFQ   002121        MP$DUP   002127        MP$MLB   000202        MP$PNT   002126
ML$INT   000000        MP$BLK   001616        MP$DVC   002114        MP$MLK   001306        MP$PRI   004411
ML$KNO   140000        MP$BLO   001452        MP$DWE   001321        MP$MLN   001631        MP$PRO   002130
ML$MON   000003        MP$BMX   002176        MP$ELT   000157        MP$MLP   006333        MP$PSS   001763
ML$SYS   000002        MP$BNP   006330        MP$ETY   001605        MP$MRB   002171        MP$PST   001762
ML$TOP   000001        MP$BRT   001620        MP$EVE   000311        MP$MRO   001605        MP$RET   001631
MN$UNL   000377        MP$BSA   002176        MP$FNC   001752        MP$MRP   006322        MP$RFA   001323
MO$ACC   000200        MP$BSD   002203        MP$GDT   002222        MP$MRT   002153        MP$RMX   002201
MO$ADD   000002        MP$BSI   002200        MP$GNM   002223        MP$MRV   006320        MP$RPA   004374
MO$ALA   000004        MP$BSP   006331        MP$GRO   002115        MP$MSB   000170        MP$RPR   001606
MO$ALI   000000        MP$BUF   000170        MP$GRP   000541        MP$MVE   000145        MP$RRT   002213
MO$CHA   000040        MP$BUP   006324        MP$GTY   002224        MP$MVI   001634        MP$RST   002212
MO$CIR   000003        MP$BUS   001643        MP$HAD   001757        MP$MVR   001751        MP$RSV   001754
MO$CLE   000100        MP$CAC   001750        MP$HBT   002142        MP$MWN   002154        MP$RSZ   001756
MO$COU   000060        MP$CAS   001762        MP$HDD   000162        MP$MXB   002172        MP$RTI   001616
MO$DAC   000000        MP$CAT   002210        MP$HTM   001612        MP$MXC   000466        MP$RTM   001611
MO$DEF   000000        MP$CCS   004406        MP$HWA   002210        MP$MXR   001630        MP$RVE   001604
MO$DPR   000000        MP$CHN   002141        MP$IAT   001322        MP$MXW   002143        MP$RVT   000156
MO$ENT   000017        MP$CIR   000144        MP$IDE   000144        MP$NAA   006335        MP$SAC   000514
MO$EVE   000100        MP$CLK   002131        MP$IDP   006327        MP$NAC   001130        MP$SAD   000543
MO$INF   000160        MP$CLN   002126        MP$IHO   000215        MP$NAP   006326        MP$SCA   000310
MO$INS   000002        MP$CLT   002211        MP$INA   002177        MP$NCT   000240        MP$SCO   000144
MO$LIN   000001        MP$CMB   002142        MP$IND   002204        MP$NET   002114        MP$SCT   002176
MO$LOG   000001        MP$CMK   000537        MP$INI   002201        MP$NLI   000765        MP$SDU   000203
MO$MIR   000031        MP$CMX   002153        MP$ITI   000776        MP$NNA   000764        MP$SDV   000160
MO$NAM   000001        MP$CND   000310        MP$LAA   000231        MP$NOD   000500        MP$SEH   004401
MO$NIC   000023        MP$CNU   001753        MP$LAN   000234        MP$NRA   006334        MP$SER   000144
MO$NOD   000000        MP$COB   000311        MP$LAR   000202        MP$NRB   000214        MP$SET   000000
MO$OFF   000001        MP$CON   002126        MP$LAS   000012        MP$NSA   001617        MP$SGZ   001644
MO$ON    000000        MP$COS   001604        MP$LBS   000203        MP$NTI   002141        MP$SID   000176
MO$OPT   000001        MP$CPF   000230        MP$LCO   000226        MP$NUM   001642        MP$SIN   000310
MO$PER   000200        MP$CPL   000232        MP$LCT   000156        MP$NVE   001274        MP$SLI   000156
MO$PRO   000002        MP$CPT   000226        MP$LHL   000232        MP$NXN=  001476        MP$SLO   000171
MO$PRS   000001        MP$CPU   000161        MP$LLE   000227        MP$OAC   000632        MP$SMX   002202
MO$REA   000200        MP$CSZ   001755        MP$LLO   004432        MP$OAN   000620        MP$SND   000500
MO$SET   000000        MP$CUS   002127        MP$LMB   002152        MP$OCO   000776        MP$SNP   006332
MO$STA   000020        MP$CVA   000540        MP$LMX   002200        MP$OHO   000214        MP$SNU   000542
MO$SUM   000000        MP$DAL   002570        MP$LNA   000144        MP$ONA   000764        MP$SOB   000524
MO$TSK   000000        MP$DCO=  001464        MP$LNO   000233        MP$ONR   002114        MP$SPA   000157
MO$VOL   000000        MP$DDT   002177        MP$LOA   000170        MP$OQL   001453        MP$SPR   000536
MO$ZER   000000        MP$DEL   001131        MP$LOG   004514        MP$OTI   000777        MP$SPS   000513
MP$ACB   000012        MP$DES   000156        MP$LOO   000620        MP$OTY   001022        MP$STA   000000
MP$ACC   000514        MP$DEV   002114        MP$LTM   001613        MP$OUS   000777        MP$STI   002140
MP$ACT   004526        MP$DFA   001320        MP$LTY   004533        MP$OVE   001010        MP$STT   002201
MP$ADD   000766        MP$DHO=  001465        MP$LWI   000230        MP$OWN   004374        MP$STY   000175
MP$ADJ   001440        MP$DIA   000173        MP$MAC   004420        MP$PAR   000036        MP$SUB   000001
MP$ADP   006325        MP$DLB   002571        MP$MAD   001630        MP$PAS   000513        MP$SUR   000156
MP$ADS   004406        MP$DLG   004521        MP$MAP   006323        MP$PCH   002152        MP$SUS   000512
MP$ALB   000036        MP$DLI=  001466        MP$MAR   001635        MP$PCO   000024        MP$SVR   000163
MP$AMC   001640        MP$DLT   002200        MP$MAV   006321        MP$PCT   000144        MP$SWI   002260
```

```
                                                      H 15
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04  Page 8-3
Symbol table
```

```
    324                                         ;
    325                                         ; CLEAR MULTIPLE ALIASES (KNOWN, ACTIVE OR ALL)
    326                                         ;
    327 000446  010067  000000G          25$:      MOV     R0,$CRUNM               ;; SET UP CURRENT UNMAPPED POINTER
    328 000452  016767  000000G 000000G  30$:      MOV     $CRUNM,$PRUNM           ;; MOVE CURRENT TO PREVIOUS
    329 000460  011067  000000G                    MOV     (R0),$CRUNM             ;; MOVE ON TO NEXT
    330 000464  011046                             MOV     (R0),-(SP)              ;; SET UP TO MAP TO ANB
    331 000466                                     CALL    @CEACC                  ;; MAP TO ANB
    332 000472  012601                             MOV     (SP)+,R1                ;; GET ANB VIRTUAL ADDRESS
    333 000474  001414                             BEQ     50$                     ;; IF EQ, FINISHED
    334 000476  126427  000011  177775             CMPB    A$TYP(R4),#MF$ALL       ;; IS THIS A CLEAR ALL OPERATION ?
    335 000504  001404                             BEQ     40$                     ;; IF EQ, YES - SKIP SCOPE CHECK
    336 000506  026164  000010  000020             CMP     A.UCB(R1),A$SCO(R4)     ;; ELSE, IS THIS ONE TO BE REMOVED ?
    337 000514  001002                             BNE     45$                     ;; IF NOT EQ, NO, KEEP LOOKING
    338 000516                           40$:      CALL    DEAANB                  ;; UNLINK AND DEALLOCATE THE NAME BLOCK
    339 000522  010100                   45$:      MOV     R1,R0                   ;; ELSE, POINT TO NEXT BLOCK
    340 000524  000752                             BR      30$                     ;; AND CONTINUE
    341 000526                           50$:      RESMAP                          ;; RESTORE MAPPING
    342
    343                                            .IF DF  R$$PRO                  ;; PRO/DECNET
    344
    345                                            MOV     @DECPT,R1               ;; POINT TO DECNET HOME BLOCK
    346                                            BIC     #DF.AUP,D$STAT(R1)      ;; UNLOCK NT.CFG PARTITION
    347
    348                                            .ENDC
    349
    350 000532                           60$:      RETURN                          ;; BACK TO USER STATE AND RETURN
```

```
                .SBTTL  FCSERR  - Input File FCS Error Handler
        ;+
        ;                               *** FCSERR ***
        ;
        ; Function:     To log FCS errors as DECnet events.  No file handling is done
        ;               to correct the error condition.
        ;
        ; Call Format:  CALL    FCSERR
        ;
        ; Inputs:       None
        ;
        ; Outputs:      R3 = load halt status
        ;               FCS error is logged to event sink
        ;               Current load sequence is marked for a halt
        ;
        ; Registers:    R0 through R4 are destroyed
        ;
        ;-

        FCSERR: MOV     #FDB,R0                 ; Get address of input file's FDB
                CMPB    F.ERR(R0),#IE.EOF       ; Did end of file occur?
                BNE     10$                     ; Branch if so, no real error
                CLR     R3                      ; Get "end of data" halt reason
                BR      20$                     ; Branch to common exit

        10$:    MOV     #FILERR,R2              ; Get $EDMSG argument block address
                MOVB    F.ERR(R0),R1            ; Get FCS error code with sign extend
                MOV     R1,(R2)                 ; Move error to argument block
                MOV     #WORKSP,R0              ; Get address of output string
                MOV     #FIOERR,R1              ; Get $EDMSG instruction string address
                CALL    $EDMSG                  ; Format the output string
                CLRB    (R0)                    ; Make it an ASCIZ string
                MOV     #WORKSP,R2              ; Recover error-specific string address
                CALL    LOGEVT                  ; Log the event
                MOV     #CS.FTL,R3              ; Get "fatal error" status

        20$:    SEC                             ; Set C-bit for error
                RETURN                          ; Exit
```

```
                                                                   H 2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL   VALUE              REFERENCES

EV$XGW  = 034111           #6-83
EV$XMX  = 000514           #6-83
EV$XRS  = 000512           #6-83
EV$XSC  = 000513           #6-83
EV$X2S  = 013500           #6-83
EV.CCB  = 000001           #6-83
EV.CIR  = 000020           #6-83
EV.LCB  = 000100           #6-83
EV.LIN  = 000004           #6-83
EV.MAP  = 000002           #6-83
EV.MOD  = 000040           #6-83
EV.NOD  = 000010           #6-83
EV.PRT  = 000200           #6-83
EXEVCL  = 000017           #8-247
EXEVEC    000566 R         #8-227        8-247
E$DATA    000020           #6-83         11-606
E$EVTS    000000           #6-83
E$LCN     000016           #6-83
E$LIN     000000           #6-83
E$MOD     000012           #6-83
E$NOD     000010           #6-83
E$PORT    000014           #6-83
E$PRM     000002           #6-83
E$STAT    000006           #6-83
E$TCB     000004           #6-83
E.CTL     000020           #6-83
E.DATA    000046           #6-83
E.EVT     000002           #6-83
E.LCN     000042           #6-83
E.LEN     000216           #6-83
E.LIN     000024           #6-83
E.LNK     000000           #6-83
E.MOD     000036           #6-83
E.NOD     000034           #6-83
E.PDV     000021           #6-83
E.PORT    000040           #6-83
E.PRM     000026           #6-83
E.PVC     000044           #6-83
E.SIZ     000022           #6-83
E.TIME    000004           #6-83
FC.MLD  = 000026            9-335        9-364        9-447        9-463        12-678
FR$BCC  = 000007           #6-83
FR$CCF  = 000001           #6-83
FR$CDF  = 000002           #6-83
FR$DAO  = 000011           #6-83
FR$EXC  = 000000           #6-83
FR$FRM  = 000010           #6-83
FR$FTL  = 000005           #6-83
FR$OPN  = 000004           #6-83
FR$RFD  = 000006           #6-83
FR$SBU  = 000012           #6-83
FR$SHO  = 000003           #6-83
```

```
                                                                   I 2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 4
SYMBOL CROSS REFERENCE                                   CREF    04.00
```

```

                         .IF DF R$$PRO     ; PRO/DECnet

                         ZTIM2:: .WORD    $ZTIME+2

                         .IFF     ; DF R$$PRO

000200  000000G          ZTIM2:: .WORD    $ZTIM2

                         .ENDC    ; DF R$$PRO

000202  000000G          CALLX:: .WORD    $CALLX

                         .IF      DF      R$$MPL

                         NCPU::  .WORD    $NCPU               ; M+ SPECIFIC VECTORS
                         SAHPT:: .WORD    $SAHPT
                         SAHDB:: .WORD    $SAHDB
                         TICLR:: .WORD    $TICLR

                         .ENDC

        000102           NMVVCL  == .-NMVVEC/2                ; VECTOR TABLE LENGTH
                         ;
                         ; WINDOW BLOCK OFFSETS
                         ;
000000                           .ASECT
        000000                   .=0
000000                   W$ACP:: .BLKW    1                   ; TCB OF ACP
000002                   W$CTX:: .BLKW    1                   ; ADDRESS OF CONTEXT AREA IN ACP
000004                   W$OPR:: .BLKW    1                   ; ADDRESS OF "OPERATE" ROUTINE
000006                   W$SIZ::                              ; LENGTH OF WINDOW BLOCK

                         ;
                         ; CONTEXT AREA OFFSETS
                         ;
        000000                   .=0
000000                   C$LINK::.BLKW    1                   ; LINK WORD
000002                   C$FLAG::.BLKW    1                   ; FLAGS
        000001             CF$BSY==1                          ; ... CONTEXT IS IN USE
        000002             CF$FDN==2                          ; ... CONTEXT FORK CAN BE RESUMED
        000004             CF$VR2==4                          ; ... NICE REQUEST FROM V2.0 NCP
        000010             CF$VR3==10                         ; ... NICE REQUEST FROM V3.0 NCP
000004                   C$SP::  .BLKW    1                   ; SAVED SP
000006                   C$STAT::.BLKW    2                   ; STATUS BLOCK (FOR FORK COMPLETION)
000012                   C$CCB::                              ; CCB ADDRESS FROM PROCESS REQUEST
000012                   C$LOC:: .BLKB    MAXLID              ; LOCAL STORAGE (NOT OVERWRITTEN BY UPDCTX)
000032                   C$DATA::.BLKB    CTXSIZ              ; USER DATA AREA
000520                   C$SIZE::                             ; LENGTH OF CONTEXT AREA

                         ;
                         ; FUNCTION DESCRIPTOR OFFSETS
                         ;
        000000                   .=0
000000                   R$FUNC::.BLKW    1                   ; FUNCTION CODE
000002                   R$FLAG::.BLKB    1                   ; FLAGS
        000200             RF$END==200                        ; ... LAST ENTRY IN TABLE
```

```
                                                                 H  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-1
Symbol table

AALOCB  000010RG     A.REL   000000 G     CF$FDN= 000002 G     CS.ERR= 100000       C.XACP  000004
AB.NPV= 000001       A.RES = 000032       CF$VR2= 000004 G     CS.FTL= 001000       C.XID   000035
AB.TYP= 000002       A.SBUF  000022 G     CF$VR3= 000010 G     CS.HCR= 000001       C.XLEN  000044
AF.AST= 000010       A.SLEN  000024 G     CF.CHN= 000001       CS.HFE= 002000       C.XPLI  000040
AF.ESQ= 000020       A.SMAP  000020 G     CF.EOM= 000004       CS.LST= 040000       C.XPT   000034
AF.LCK= 000040       A.STA   000015 G     CF.HDR= 000020       CS.MTL= 004000       C.XSVC  000042
AF.MDE= 000200       A.TRAN  000022 G     CF.LB = 100000       CS.RNG= 000010       C.XTC   000037
AF.NOT= 000002       BIT0  = 000001 G     CF.LIN= 000002       CS.ROV= 000004       C.X25   000036
AF.OOB= 000004       BIT1  = 000002 G     CF.SOM= 000010       CS.RSN= 010000       DAOERR  002242R
AF.QUE= 000100       BIT10 = 002000 G     CF.SYN= 000040       CS.SHU= 000001       DDEACB  000016RG
AF.XCC= 000001       BIT11 = 004000 G     CF.TRN= 000100       CS.SID= 000002       DDMAN   000066RG
AK.BUF= 000200       BIT12 = 010000 G     CM$RSX= 000200       CS.STR= 000004       DEACTX= ****** GX
AK.GBI= 000202       BIT13 = 020000 G     CM.CDS= 000004       CS.SUC= 000001       DECPT   000070RG
AK.GGF= 000303       BIT14 = 040000 G     CM.CEN= 000003       CS.TMO= 020000       DELFLG  000216R
AK.OCB= 000201       BIT15 = 100000 G     CM.CIR= 000002       CS.XUR= 000004       DREXT   000020RG
ALLCTX= ****** GX    BIT2  = 000004 G     CM.ELM= 000005       CTXHD   000232RG     D$$BUG= 177514
ALNERR  002234R      BIT3  = 000010 G     CM.EXT= 000006       CTXSIZ= 000466 G     D$$ISK= 000000
AS.CAA= 000006       BIT4  = 000020 G     CM.FMT= 100000       CURCTX  000212RG     D$$L11= 000001
AS.DIS= 000002       BIT5  = 000040 G     CM.HRD= 000002       CURIOP  000210RG     D$$YNC= 000000
AS.DLT= 000001       BIT6  = 000100 G     CM.IND= 000002       C$CCB   000012 G     D$$YNM= 000000
AS.FPA= 000001       BIT7  = 000200 G     CM.INE= 000001       C$DATA  000032 G     EXRQN   000022RG
AS.PFA= 000004       BIT8  = 000400 G     CM.LIN= 000000       C$FLAG  000002 G     EXRQP   000024RG
AS.RCA= 000002       BIT9  = 001000 G     CM.LKT= 000007       C$LINK  000000 G     E$$XPR= 000000
AS.REA= 000005       BLXIO   000012RG     CM.LOO= 000001       C$LOC   000012 G     FC.CCP= 000020
AS.RRA= 000003       BUFFER  000734RG     CM.MSG= 000011       C$SIZE  000520 G     FC.CTL= 000006
A$$CHK= 000000       BUFSIZ= 000454       CM.RMT= 000010       C$SP    000004 G     FC.KCP= 000016
A$$CPS= 000000       CALLX   000202RG     CM.XLO= 000004       C$STAT  000006 G     FC.KIL= 000004
A$$PRI= 000000       CB.CCB= 000002       CONTEX  000244RG     C$$CKP= 000000       FC.MAN= 000024
A$$TRP= 000000       CB.DDM= 000040       COPT    000014RG     C$$ORE= 000400       FC.MLD= 000026
A.ACC   000014 G     CB.DLC= 000020       CP.DCF= 000040       C$$RSH= 177564       FC.PCT= 000030
A.ACCE  000000 G     CB.RDB= 000004       CP.DSB= 000010       C.ADD   000034       FC.PWR= 000022
A.AST   000006 G     CB.SDB= 000010       CP.EXT= 000400       C.BID   000003       FC.RCE= 000002
A.BYT   000004 G     CB.SLI= 000100       CP.HDL= 000007       C.BUF   000014       FC.RCP= 000014
A.CALL  000024 G     CB.XLB= 000001       CP.LGO= 000004       C.BUF1  000014       FC.TIM= 000010
A.CBL   000002 G     CCBAF   000046RG     CP.MSG= 000002       C.BUF2  000024       FC.XCP= 000012
A.CONN  000012 G     CCBAL   000050RG     CP.NIO= 000100       C.CNT   000020       FC.XME= 000000
A.DEQU  000002 G     CCBCT   000052RG     CP.NUL= 000001       C.CNT1  000020       FDNCNT  000224RG
A.DIS   000002 G     CCBGT   000054RG     CP.PRV= 000020       C.CNT2  000030       FE.CEX= ****** GX
A.DISC  000014 G     CCBNM   000060RG     CP.PS = 177400       C.FLG   000022       FILHD   000072RG
A.DQSR  177776 G     CCBRT   000056RG     CP.PSI= 000200       C.FLG1  000022       FINREQ  002736R
A.FLEN  000011 G     CC.LLC= 000200       CP.RST= 000200       C.FLG2  000032       FMASK   000026RG
A.IBUF  000014 G     CEACC   000062RG     CP.SGL= 000040       C.FNC   000010       FRKDON= ****** GX
A.ILEN  000016 G     CE.ABO= 100362       CP.XCF= 000100       C.LIN   000006       FS.AST= 000000
A.IMAP  000012 G     CE.DAO= 100346       CP.2FR= 000030       C.LNK   000000       FS.CIB= 002000
A.INPU  000006 G     CE.DIS= 100366       CSBRT   000064RG     C.MOD   000011       FS.CRA= 001000
A.IOS   000026 G     CE.ERR= 100370       CS.ABO= 000100       C.NSP   000004       FS.DIS= 013000
A.KSR5  177774 G     CE.ILN= 100350       CS.BRO= 000002       C.PCPL  000011 G     FS.DVC= 001000
A.LIN   000012 G     CE.LTO= 100356       CS.BUF= 000200       C.PDPL  000010 G     FS.ENB= 012000
A.MAS   000004 G     CE.MOP= 100372       CS.CES= 000002       C.PNAM  000002 G     FS.EXI= 001000
A.NPR   000010 G     CE.NTE= 100361       CS.CHN= 000010       C.PRMT  000012 G     FS.GET= 006000
A.NUM   000010 G     CE.RTE= 100376       CS.CMP= 000200       C.PRO   000042       FS.HLT= 000000
A.OUTP  000010 G     CE.SRC= 100364       CS.DCR= 000400       C.PSTS  000006 G     FS.INI= 000000
A.PLEN= 000016       CE.STP= 100352       CS.DEF= 000004       C.PTCB  000000 G     FS.KIL= 000000
A.POWE  000004 G     CE.TME= 100354       CS.DEV= 000002       C.RSV   000002       FS.LCL= 100000
A.PRM   000012 G     CE.TMO= 100374       CS.DIS= 000040       C.STA   000007       FS.LTM= 001000
A.PROC  000020 G     CE.UNS= 100344       CS.ENA= 000001       C.STS   000012       FS.MNT= 004000
A.RECE  000016 G     CF$BSY= 000001 G     CS.ENB= 000020       C.URM   177776       FS.MSN= 014000
```

```
                                                                 I  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-2
Symbol table
```

Table of contents

```
     5-   53    MACRO CALLS AND LOCAL DATA
     6-   59    NTLAST - NTL COMPLETION AST HANDLING
     7-   94    DLXAST - DLX QIO COMPLETION AST HANDLING
     8-  120    MRKAST -MARK TIME COMPLETION AST HANDLING
```

```
                                        .SBTTL   CONTEXT AREA POOL AND INITIALIZATION CODE
                                ;+
                                ;**-INIBUF-INITIALIZE ACP BUFFERS AND RELEASE CODE TO CONTEXT AREA POOL
                                ;
                                ; THIS ROUTINE IS CALLED TO INITIALIZE THE CONTEXT POOL POINTER AND
                                ; LENGTH LOCATIONS. IF ANY I/O PACKETS ARE FOUND IN THE RECEIVE QUEUE
                                ; WITH THE "BEING PROCESSED" FLAG SET (LOW BIT OF I.TCB), THEY ARE
                                ; COMPLETED WITH AND ERROR OF "IO.ABO". ANY BUFFERS IN THE NETWORK
                                ; MANAGEMENT COMPLETION LIST ARE RELEASED (FOR VOLATILE ACP ONLY).
                                ;
                                ; INPUTS:
                                ;       NONE
                                ;
                                ; OUTPUTS:
                                ;       IF CC, SUCCESS, AND
                                ;          POOLPT = POINTER TO START OF CONTEXT AREA POOL
                                ;          POOLSZ = INITIAL SIZE OF CONTEXT AREA POOL
                                ;       ELSE, INTIALIZATION FAILURE
                                ;-

000000                                  .PSECT   $$$XXX,OVR
000000                          CTXPOO:                                 ; START OF CONTEXT POOL
000000                                  .BLKW    16.                    ; GET TASK BUFFER

000040                          INIBUF::                                ; INITIALIZE ACP (ONCE ONLY CODE)
000040                                  SWSTK$   30$                    ;; ENTER SYSTEM STATE
000044  017705  000000G         5$:     MOV      @TKTCB,R5              ;; POINT TO OUR TCB              ; RJK02
000050  062705  000000G                 ADD      #T.RCVL,R5             ;; ... AND TO OUR RECEIVE QUEUE
000054  010504                          MOV      R5,R4                  ;; ...
000056  011403                  10$:    MOV      (R4),R3                ;; GET NEXT PACKET ADDRESS
000060  001422                          BEQ      20$                    ;; IF EQ, NONE LEFT
000062  032763  000001  000004          BIT      #1,I.TCB(R3)           ;; IS THE PACKET FLAGGED AS "SEEN"
000070  001002                          BNE      11$                    ;; IF NE, YES - FINISH IT IN ERROR
000072  010304                          MOV      R3,R4                  ;; ELSE, KEEP LOOKING IN LIST FROM HERE
000074  000770                          BR       10$                    ;; ...
000076  005363  000004          11$:    DEC      I.TCB(R3)              ;; CLEAR "SEEN" INDICATOR
000102  011314                          MOV      (R3),(R4)              ;; UNLINK PACKET FROM LIST
000104  001002                          BNE      12$                    ;; IF NE, NOT LAST IN LIST
000106  010465  000002                  MOV      R4,2(R5)               ;; ELSE, CLOSE UP LIST
000112  012700  000000C         12$:    MOV      #IE.ABO&377,R0         ;; RETURN ERROR OF "IE.ABO"
000116  005001                          CLR      R1                     ;; ...
000120                                  CALL     @IOFIN                 ;; FINISH I/O                    ; RJK02
000124  000747                          BR       5$                     ;; AND CONTINUE SCAN

000126                          20$:    .IF DF   MN$VOL                 ;; VOLATILE ACP ONLY
000126  032777  000000G 000000G         BIT      #FE.CEX,@FMASK         ;; COMM EXEC LOADED ?            ; RJK02
000134  001432                          BEQ      25$                    ;; IF EQ, NO - SKIP CHECKING LIST
000136  016704  000000G                 MOV      NMCLH,R4               ;; POINT TO NEXT COMPLETED BUFFER; RJK02
000142  010403                  21$:    MOV      R4,R3                  ;; SAVE UNLINK ADDRESS
000144  011404                          MOV      (R4),R4                ;; NEXT CCB
000146  001425                          BEQ      25$                    ;; IF EQ, NO BUFFERS TO RETURN
000150  026564  000000G 000024          CMP      T.NAM(R5),C.BUF2(R4)   ;; COMPARE TASK NAMES
000156  001371                          BNE      21$                    ;; IF NE, NOT ONE OF OURS
000160  026564  000002G 000026          CMP      T.NAM+2(R5),C.BUF2+2(R4) ;; COMPARE TASK NAMES
000166  001365                          BNE      21$                    ;; IF NE, NOT ONE OF OURS
000170  011413                          MOV      (R4),(R3)              ;; ELSE, UNLINK BUFFER FROM LIST
000172  001002                          BNE      22$                    ;; IF NE, NOT LAST IN LIST
```

```
**FILE**ID**MANBYE
```

```
MM      MM     AAAAAA     NN      NN   BBBBBBBB     YY      YY   EEEEEEEEEE
MM      MM     AAAAAA     NN      NN   BBBBBBBB     YY      YY   EEEEEEEEEE
MMMM  MMMM   AA      AA   NN      NN   BB      BB   YY      YY   EE
MMMM  MMMM   AA      AA   NN      NN   BB      BB   YY      YY   EE
MM  MM  MM   AA      AA   NNNN    NN   BB      BB     YY  YY     EE
MM  MM  MM   AA      AA   NNNN    NN   BB      BB     YY  YY     EE
MM      MM   AA      AA   NN  NN  NN   BBBBBBBB         YY       EEEEEEEE
MM      MM   AA      AA   NN  NN  NN   BBBBBBBB         YY       EEEEEEEE
MM      MM   AAAAAAAAAA   NN    NNNN   BB      BB       YY       EE
MM      MM   AAAAAAAAAA   NN    NNNN   BB      BB       YY       EE
MM      MM   AA      AA   NN      NN   BB      BB       YY       EE                ....
MM      MM   AA      AA   NN      NN   BB      BB       YY       EE                ....
MM      MM   AA      AA   NN      NN   BBBBBBBB         YY       EEEEEEEEEE        ....
MM      MM   AA      AA   NN      NN   BBBBBBBB         YY       EEEEEEEEEE        ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20
Table of contents

```
                                                                                H 8
MANBYE - PERFORM LOGOUT CLEANUP MACRO V05.03b Wednesday 18-Sep-85 17:20  Page 7-4
Symbol table

MX$OWN  000040        M$GROL  000053        M$SCO   000016        N$$MCP= 000001        R$$DER= 000000
MX$PAR  000036        M$GTY   000106        M$SCR   000404        N$$MLL= 000001        R$$K11= 000001
MX$PAS  000010        M$LEN   000442        M$SMX   000236        N$$MOV= 000010        R$$SND= 000000
MX$RAC  000047        M$LMX   000234        M$SND   000261        N$$NCT= 000001        R$$11M= 000000
MX$RID  000047        M$NET   000240        M$SNDL  000260        N$$PEM= 000001        SF$ANM  000001
MX$RPS  000047        M$NETL  000237        M$SPS   000353        O.AST   000006        SF$DNM  000002
MX$SID  000040        M$OBNA  000270        M$SPSL  000352        O.EFN   000010        S$$WRG= 000000
MX$SNK  000377        M$DBNL  000267        M$STA   000074        O.ESB   000012        S$$YSZ= 007600
MX$TYP  000077        M$OBNR  000112        M$SUS   000364        O.LGTH= 000034        TKTCB = ****** GX
MX$UID  000020        M$DBNU  000110        M$SUSL  000363        O.LNK   000000        T$$KMG= 000000
M$BDF   000222        M$OFLG  000000        M$TXT   000166        O.MCRL  000002        T$$MIN= 000000
M$BMX   000224        M$OPT   000012        M$WDF   000226        O.PTCB  000004        T.UCB = ****** GX
M$BUF   000014        M$PAR   000010        M$WMX   000227        O.STAT  000014        UA.ACC= 000001
M$CALL  000120        M$PCH   000172        M$$CRB= 000124        PC.ALF= 000004        UA.ALL= 000400
M$CAT   000230        M$PCHC  000170        M$$CRX= 000000        PC.ALM= 001000        UA.CAL= 000100
M$CLT   000231        M$PCHE  000222        M$$FCS= 000000        PC.HIH= 000001        UA.COM= 000200
M$CMK   000122        M$PCT   000076        M$$MGE= 000000        PC.LOW= 000002        UA.ECH= 000004
M$CVA   000142        M$PLNL  000310        M$$NET= 000000        PC.NRM= 000400        UA.PRD= 000002
M$DST   000033        M$PMC   000102        M$$OVR= 000000        PC.XAF= 000010        UA.PUT= 000040
M$DSTL  000032        M$PNL   000311        NM.CLN  003400        PC.XIT= 000200        UA.SPE= 000020
M$DTE   000022        M$PRI   000116        NM.INI  002400        PF.ALL= 177777        UA.TRA= 002000
M$DTEL  000020        M$RMX   000235        NM.OPR  003000        PF.INS= 000040        UA.TRN= 001000
M$DTEP  000100        M$RRT   000233        NM.VR2  002401        PF.LDG= 000100        UA.TYP= 000010
M$ENT   000006        M$RST   000232        NM.VR3  002402        PF.REQ= 000200        V$$CTR= 001000
M$FLAG  000002        M$SAC   000332        N$$ACC= 000001        P$$P45= 000000        X$$DBT= 000000
M$FLG2  000004        M$SACL  000331        N$$BUF= 000001        P$$WRD= 000000        $$$SYS= 004374
M$GNM   000104        M$SAD   000162        N$$LDV= 000001        Q$$OPT= 000010        $$$SYX= 000000
M$GRO   000054

. ABS.  177776    000   (RW,I,GBL,ABS,DVR)
        000100    001   (RW,I,LCL,REL,CON)
$$$BYE  000002    002   (RW,I,LCL,REL,CON)
$$$BYF  000002    003   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 232
Work  file writes: 165
Size of work file: 34838 Words  ( 137 Pages)
Size of core pool: 19272 Words  ( 74 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:31.18
DB1:MANBYE.T7,[135,134]MANBYE/-SP=DB1:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANBYE
```

```
                                                                                I 8
**FILE**ID**MANFRK
```

```
**FILE**ID**MANQIO










MM      MM     AAAAAA     NN      NN     QQQQQQ     IIIIII     000000
MM      MM     AAAAAA     NN      NN     QQQQQQ     IIIIII     000000
MMMM  MMMM   AA      AA   NN      NN   QQ      QQ       II       00      00
MMMM  MMMM   AA      AA   NN      NN   QQ      QQ       II       00      00
MM  MM  MM   AA      AA   NNNN    NN   QQ      QQ       II       00      00
MM  MM  MM   AA      AA   NNNN    NN   QQ      QQ       II       00      00
MM      MM   AA      AA   NN  NN  NN   QQ      QQ       II       00      00
MM      MM   AA      AA   NN  NN  NN   QQ      QQ       II       00      00
MM      MM   AAAAAAAAAA   NN    NNNN   QQ  QQ  QQ       II       00      00
MM      MM   AAAAAAAAAA   NN    NNNN   QQ  QQ  QQ       II       00      00
MM      MM   AA      AA   NN      NN   QQ    QQ         II       00      00   ....
MM      MM   AA      AA   NN      NN   QQ    QQ         II       00      00   ....
MM      MM   AA      AA   NN      NN     QQQQ  QQ     IIIIII     000000     ....
MM      MM   AA      AA   NN      NN     QQCQ  QQ     IIIIII     000000     ....





LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
MANQIO - ISSUE NETWORK MANAGEME MACRO V05.03b Saturday 29-Jun-85 10:30
Table of contents
```

```
ME.UCO   177770      MO$MIR  000031      MP$CMX   002153      MP$ITI   000776      MP$NNA   000764
MF$ACT   177776      MO$NAM  000001      MP$CND   000310      MP$LAA   000231      MP$NOD   000500
MF$ADD   000000      MO$NIC  000023      MP$CNU   001753      MP$LAN   000234      MP$NRA   006334
MF$ADJ   177774      MO$NOD  000000      MP$COB   000311      MP$LAR   000202      MP$NRB   000214
MF$ALL   177775      MO$OFF  000001      MP$CON   002126      MP$LAS   000012      MP$NSA   001617
MF$BYE   000300      MO$ON   000000      MP$COS   001604      MP$LBS   000203      MP$NTI   002141
MF$CHA   000023      MO$OPT  000001      MP$CPF   000230      MP$LCO   000226      MP$NUM   001642
MF$DUM   000020      MO$PER  000200      MP$CPL   000232      MP$LCT   000156      MP$NVE   001274
MF$EVT   000001      MO$PRO  000002      MP$CPT   000226      MP$LHL   000232      MP$NXN=  001476
MF$FUP   000264      MO$PRS  000001      MP$CPU   000161      MP$LLE   000227      MP$OAC   000632
MF$KNO   177777      MO$REA  000200      MP$CSZ   001755      MP$LLO   004432      MP$OAN   000620
MF$LOA   000017      MO$SET  000000      MP$CUS   002127      MP$LMB   002152      MP$OCO   000776
MF$LOO   177775      MO$STA  000020      MP$CVA   000540      MP$LMX   002200      MP$OHO   000214
MF$REA   000024      MO$SUM  000000      MP$DAL   002570      MP$LNA   000144      MP$ONA   000764
MF$SIG   177773      MO$TSK  000000      MP$DCO=  001464      MP$LNO   000233      MP$ONR   002114
MF$SPF   000302      MO$VOL  000000      MP$DDT   002177      MP$LOA   000170      MP$OQL   001453
MF$SYS   000026      MO$ZER  000000      MP$DEL   001131      MP$LOG   004514      MP$OTI   000777
MF$TES   000022      MP$ACB  000012      MP$DES   000156      MP$LOO   000620      MP$OTY   001022
MF$TRA   000301      MP$ACC  000514      MP$DEV   002114      MP$LTM   001613      MP$OUS   000777
MF$TRI   000021      MP$ACT  004526      MP$DFA   001320      MP$LTY   004533      MP$OVE   001010
MF$ZER   000025      MP$ADD  000766      MP$DHO=  001465      MP$LWI   000230      MP$OWN   004374
ML$ALL   100000      MP$ADJ  001440      MP$DIA   000173      MP$MAC   004420      MP$PAR   000036
ML$CLS   000000      MP$ADP  006325      MP$DLB   002571      MP$MAD   001630      MP$PAS   000513
ML$CON   000001      MP$ADS  004406      MP$DLG   004521      MP$MAP   006323      MP$PCH   002152
ML$EXT   000001      MP$ALB  000036      MP$DLI=  001466      MP$MAR   001635      MP$PCO   000024
ML$FIL   000002      MP$AMC  001640      MP$DLT   002200      MP$MAV   006321      MP$PCT   000144
ML$FIR   000000      MP$AMH  001641      MP$DRO   001441      MP$MBN   001636      MP$PDT   002114
ML$INT   000000      MP$ANB  004420      MP$DST   000454      MP$MBR   001637      MP$PHA   000170
ML$KNO   140000      MP$ASB  000024      MP$DTE   002140      MP$MBU   001642      MP$PHY   000012
ML$MON   000003      MP$ASC  004432      MP$DTY=  001452      MP$MCB   000156      MP$PLI   000025
ML$SYS   000002      MP$AUS  000512      MP$DUA   000207      MP$MCO   001632      MP$PLN   002140
ML$TOP   000001      MP$BBT  002165      MP$DUC   000210      MP$MDE   004421      MP$PLO   000012
MN$UNL   000377      MP$BDF  002164      MP$DUM   000202      MP$MHO   001633      MP$PMC   004374
MOCHIN   000552RG    MP$BFQ  002121      MP$DUP   002127      MP$MLB   000202      MP$PNT   002126
MOCHOP   000722RG    MP$BLK  001616      MP$DVC   002114      MP$MLK   001306      MP$PRI   004411
MOID     000572R     MP$BLO  001452      MP$DWE   001321      MP$MLN   001631      MP$PRO   002130
MOREIN   000560RG    MP$BMX  002176      MP$ELT   000157      MP$MLP   006333      MP$PSS   001763
MOREOP   000722RG    MP$BNP  006330      MP$ETY   001605      MP$MRB   002171      MP$PST   001762
MOZEIN   000566RG    MP$BRT  001620      MP$EVE   000311      MP$MRO   001605      MP$RET   001631
MOZEOP   000722RG    MP$BSA  002176      MP$FNC   001752      MP$MRP   006322      MP$RFA   001323
MO$ACC   000200      MP$BSD  002203      MP$GDT   002222      MP$MRT   002153      MP$RMX   002201
MO$ADD   000002      MP$BSI  002200      MP$GNM   002223      MP$MRV   006320      MP$RPA   004374
MO$ALA   000004      MP$BSP  006331      MP$GRO   002115      MP$MSB   000170      MP$RPR   001606
MO$ALI   000000      MP$BUF  000170      MP$GRP   000541      MP$MVE   000145      MP$RRT   002213
MO$CHA   000040      MP$BUP  006324      MP$GTY   002224      MP$MVI   001634      MP$RST   002212
MO$CIR   000003      MP$BUS  001643      MP$HAD   001757      MP$MVR   001751      MP$RSV   001754
MO$CLE   000100      MP$CAC  001750      MP$HBT   002142      MP$MWN   002154      MP$RSZ   001756
MO$COU   000060      MP$CAS  001762      MP$HDD   000162      MP$MXB   002172      MP$RTI   001616
MO$DAC   000000      MP$CAT  002210      MP$HTM   001612      MP$MXC   000466      MP$RTM   001611
MO$DEF   000000      MP$CCS  004406      MP$HWA   002210      MP$MXR   001630      MP$RVE   001604
MO$DPR   000000      MP$CHN  002141      MP$IAT   001322      MP$MXW   002143      MP$RVT   000156
MO$ENT   000017      MP$CIR  000144      MP$IDE   000144      MP$NAA   006335      MP$SAC   000514
MO$EVE   000100      MP$CLK  002131      MP$IDP   006327      MP$NAC   001130      MP$SAD   000543
MO$INF   000160      MP$CLN  002126      MP$IHO   000215      MP$NAP   006326      MP$SCA   000310
MO$INS   000002      MP$CLT  002211      MP$INA   002177      MP$NCT   000240      MP$SCO   000144
MO$LIN   000001      MP$CMB  002142      MP$IND   002204      MP$NET   002114      MP$SCT   002176
MO$LOG   000001      MP$CMK  000537      MP$INI   002201      MP$NLI   000765      MP$SDU   000203
```

```
                                                      H 11
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31  Page 8
TRAALI - TRANSLATE AN ALIAS TO AN NAME

                                        .SBTTL  TRAALI - TRANSLATE AN ALIAS TO AN NAME
                                ;+
                                ; **-TRAALI-TRANSLATE AN ALIAS TO A NAME
                                ;
                                ; THIS ROUTINE TRANSLATES AN ALIAS TO A NAME
                                ;
                                ; INPUTS:
                                ;       R1 = POINTER TO ALIAS
                                ;       R3 = ADDRESS OF BUFFER
                                ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
                                ;
                                ; OUTPUTS:
                                ;       R0 = RETURN STATUS
                                ;       R2 = LENGTH OF DATA TO RETURN (0 OR 6)
                                ;-

000134                          TRAALI::
000134  005002                          CLR     R2                      ; ASSUME NO NAME TO RETURN
000136  016705  000000G                 MOV     CURIOP,R5               ; GET ADDRESS OF I/O PACKET
000142  026527  000006G 000006          CMP     I.PRM+6(R5),#6          ; WAS A SIX BYTE ALIAS SUPPLIED ?
000150  001403                          BEQ     10$                     ; IF EQ, YES - TRY TO MAP IT
000152  012700  177776                  MOV     #ME.FOR,R0              ; ELSE, MESSAGE FORMAT ERROR
000156  000416                          BR      30$                     ; AND RETURN
000160  010100                  10$:    MOV     R1,R0                   ; POINT TO THE ALIAS
000162                                  CALL    MAPALI                  ; MAP THE ALIAS TO A NAME
000166  103003                          BCC     20$                     ; IF CC, SUCCESS
000170  012700  177770                  MOV     #ME.UCO,R0              ; ELSE ''UNKNOWN COMPONENT''
000174  000407                          BR      30$                     ; ...
000176  012023                  20$:    MOV     (R0)+,(R3)+             ; SET THE NAME IN THE BUFFER
000200  012023                          MOV     (R0)+,(R3)+             ; ...
000202  011013                          MOV     (R0),(R3)               ; ...
000204  012700  000001                  MOV     #MS.SUC,R0              ; SUCCESS !
000210  012702  000006                  MOV     #6,R2                   ; SIX BYTES TO RETURN
000214                          30$:    RETURN
```

```
                                                      I 11
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31  Page 9
TRAACC - TRANSLATE AN ALIAS TO AN NAME AND ACCESS CONTROL
```

```
                                                .SBTTL  INIREQ - INITIALIZE ACP FOR A REQUEST
                                        ;+
                                        ; **-INIREQ-INITIALIZE ACP FOR A REQUEST
                                        ;
                                        ; THIS ROUTINE IS CALLED TO SET THE CORRECT ENVIRONMENT FOR DISPATCHING
                                        ; TO AN INITIAIZE ROUTINE. IT ASSIGNS THE MAILBOX LUN TO NS:, AND
                                        ; CHECKS TO SEE IF THE COMM EXEC IS LOADED. IF THE COMM EXEC IS NOT
                                        ; LOADED, AN ERROR IS RETURNED (FOR ALL FUNCTIONS EXEPT "SET SYSTEM").
                                        ;
                                        ; INPUTS:
                                        ;       R3 = BUFFER POINTER
                                        ;       R5 = FUNCTION TABLE ENTRY
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CC, VALID TO DISPATCH TO INITIALIZE FUNCTION
                                        ;       IF CS, ERROR RETURN AND:
                                        ;          R0 = ERROR CODE
                                        ;          R2 = LENGTH OF DATA IN BUFFER
                                        ;-

000014                                  INIREQ::
000014  032777  020000  000000G                 BIT     #FE.CEX,@FMASK          ; IS THE COMM EXEC LOADED ?     ; RJK01
000022  001425                                  BEQ     40$                     ; IF EQ, NO - CHECK FOR SET SYSTEM

000024                                          SWSTK$  30$                     ;; ENTER SYSTEM STATE
000030  012702  004640                          MOV     #^RAUX,R2               ;; GET AUX PDV NAME
000034                                          CALL    @PDVID                  ;; LOOK FOR AUX                 ; RJK01
000040  103406                                  BCS     10$                     ;; BR IF NOT FOUND
000042  067702  000000G                         ADD     @PDVTA,R2               ;; INDEX INTO PDV VECTOR TABLE  ; RJK01
000046  011202                                  MOV     (R2),R2                 ;; GET PDV ADDRESS
000050  005762  000012                          TST     Z.PCB(R2)               ;; IS IT LOADED YET?
000054  001005                                  BNE     20$                     ;; BR IF YES
000056                                  10$:    RETC    R0                      ;; ELSE SET USER C-BIT
000070                                  20$:    RETURN                          ;; LEAVE SYSTEM STATE

000072  103404                          30$:    BCS     50$                     ; BR IF AUX NOT LOADED
000074  000415                                  BR      60$                     ; ELSE CONTINUE
000076  021527                          40$:    CMP     (R5),(PC)+              ; IS THE FUNCTION A "SET SYSTEM" ?
000100     223     006                          .BYTE   MF$CHA!CM$RSX,MO$SET!ME$SYS
000102  001412                                  BEQ     60$                     ; IF EQ, YES - PROCEED
000104  112700  177765                  50$:    MOVB    #ME.CST,R0              ; ELSE, SET COMPONENT STATE ERROR CODE
000110  110023                                  MOVB    R0,(R3)+                ; ... PLACE IT IN THE BUFFER
000112  112723  000006                          MOVB    #ME$SYS,(R3)+           ; SET ERROR DETAIL TO SYSTEM ENTITY
000116  105013                                  CLRB    (R3)                    ; ...
000120  012702  000003                          MOV     #3,R2                   ; SET LENGTH OF RETURN TO 3 BYTES
000124  000261                                  SEC                             ; INDICATE ERROR
000126  000411                                  BR      70$                     ; AND RETURN
000130                                  60$:    ALUN$S  #$MBLUN,#"NS            ; TRY TO ASSIGN LUN TO NS:
000150  000241                                  CLC                             ; IGNORE ERROR (MAYBE NO DECNET)
000152                                  70$:    RETURN

        000001                                  .END
```

```
ALCHIN= ****** GX       MB$ENA  000000        MC$NPR  001630        MD$DMP  000022        ME$EXE  000200
ALCHOP= ****** GX       MB$FUL  000002        MC$NRE  001200        MD$DMR  000050        ME$LIN  000001
ALREIN= ****** GX       MB$MIX  000002        MC$NRT  001166        MD$DMV  000042        ME$LOG  000002
ALREOP= ****** GX       MB$ONE  000001        MC$NVR  001642        MD$DN   000016        ME$MOD  000004
ARREIN= ****** GX       MB$RXO  000001        MC$P08  000000        MD$DP   000000        ME$NOD  000000
ARREOP= ****** GX       MB$TXO  000000        MC$P11  000001        MD$DPV  000044        ME$NON  177777
A$$CHK= 000000          MB$ZER  000000        MC$RFL  002046        MD$DQ   000006        ME$OBJ  000007
A$$CPS= 000000          MC$BAB  006073        MC$SBU  002051        MD$DTE  000024        ME$OB2  000004
A$$PRI= 000000          MC$BID  001765        MC$SFL  002044        MD$DU   000002        ME$PRO  000005
A$$TRP= 000000          MC$BMC  001767        MC$SYC  000012        MD$DUP  000012        ME$SYS  000006
CICHIN= ****** GX       MC$BSC  001766        MC$SYL  000036        MD$DV   000026        ME.BLO  177744
CICHOP= ****** GX       MC$CAC  001442        MC$SYR  000050        MD$DZ   000030        ME.CON  177753
CIREIN= ****** GX       MC$CAP  001440        MC$SYS  000024        MD$FUL  000000        ME.CST  177765
CIREOP= ****** GX       MC$CCL  001445        MC$SYZ  000000        MD$HAL  000001        ME.DIS  177755
CIZEIN= ****** GX       MC$CDC  002045        MC$TBR  001140        MD$HEL  000002        ME.DON  177600
CIZEOP= ****** GX       MC$CDP  001441        MC$TBS  001141        MD$KCP  000013        ME.FCO  177762
CM$RSX= 000200          MC$CIF  001465        MC$TDR  002053        MD$KDP  000034        ME.FIO  177756
C$$CKP= 000000          MC$CLD  001464        MC$TIM  000000        MD$KDZ  000036        ME.FOP  177763
C$$ORE= 000400          MC$COU  100000        MC$TYP  007777        MD$KL   000040        ME.FOR  177776
C$$RSH= 177564          MC$CTL  001454        MC$UBU  002052        MD$KMX  000054        ME.FUN  177777
D$$BUG= 177514          MC$CTR  001452        MC$UFD  002047        MD$KMY  000052        ME.GRO  177745
D$$ISK= 000000          MC$CTS  001453        MC$UMR  001132        MD$LEN  000050        ME.HAR  177750
D$$L11= 000001          MC$DDV  002050        MC$UMS  001133        MD$PCL  000011        ME.IID  177767
D$$YNC= 000000          MC$LBR  001750        MC$VAX  000003        MD$QNA  000005        ME.LCO  177766
D$$YNM= 000000          MC$LBS  001751        MC$WID  060000        MD$UNA  000001        ME.LPR  177757
E$$XPR= 000000          MC$LDI  001774        MC$WIH  040000        MD$WIT  000002        ME.MPR  177773
FTABLE  000000RG    002 MC$LDO  001775        MC$WIL  020000        MD.ABO  000017        ME.MVE  177771
F$$LVL= 000001          MC$LDR  001762        MC$W08  020000        MD.ACC  000010        ME.OPE  177747
G$$TPP= 000000          MC$LDS  001763        MC$W16  040000        MD.ADB  000016        ME.PLO  177751
G$$TSS= 000000          MC$LLB  002021        MC$W32  060000        MD.BOB  000011        ME.PMI  177743
G$$TTK= 000000          MC$LLP  002115        MC$XBR  001750        MD.DIA  000007        ME.PNA  177752
G$$WRD= 000000          MC$LLR  002007        MC$XBS  001751        MD.DOB  000015        ME.PRI  177775
I$$RAR= 000000          MC$LOO  000001        MC$XCJ  000322        MD.DUM  000002        ME.PTY  177772
I$$RDN= 000000          MC$LRB  002020        MC$XCR  002260        MD.FAI  000014        ME.PVA  177760
K$$CNT= 177546          MC$LRP  002114        MC$XCS  002261        MD.FNA  000001        ME.RES  177761
K$$CSR= 177546          MC$LRR  002006        MC$XDR  001762        MD.FOB  000006        ME.ROO  177754
K$$LDC= 000000          MC$LSI  002032        MC$XDS  001763        MD.LOA  000001        ME.SIZ  177774
K$$TPS= 000074          MC$LST  002033        MC$XFR  002272        MD.LSH  000020        ME.SYS  177746
LC$EXT  000000          MC$LUP  006072        MC$XFS  002273        MD.NOB  000012        ME.UCO  177770
LC$INT  000001          MC$MAP  010000        MC$XLJ  000323        MD.NON  177777        MF$ACT  177776
LO$LP = 000000          MC$MBL  001764        MC$XLR  002330        MD.PER  000000        MF$ADD  000000
LICHIN= ****** GX       MC$MBX  004420        MC$XMA  000310        MD.RES  000004        MF$ADJ  177774
LICHOP= ****** GX       MC$MBY  001752        MC$XMC  002305        MD.ROB  000005        MF$ALL  177775
LIREIN= ****** GX       MC$MPX  004406        MC$XMS  002304        MD.RSH  000013        MF$BYE  000300
LIREOP= ****** GX       MC$NAP  001604        MC$XNR  002332        MD.SDU  000005        MF$CHA  000023
LIZEIN= ****** GX       MC$NBR  001130        MC$XRC  002316        MD.SLD  000003        MF$DUM  000020
LIZEOP= ****** GX       MC$NBS  001131        MC$XRR  002331        MD.SNA  000000        MF$EVT  000001
LOCHIN= ****** GX       MC$NCR  001154        MC$XRS  002342        MD.TLO  000004        MF$FUP  000264
LOCHOP= ****** GX       MC$NCS  001155        MC$020  000002        MD.UNA  000002        MF$KNO  177777
LOGOUT= ****** GX       MC$NML  001274        MD$CI   000007        MD.UNR  000003        MF$LDA  000017
LOREIN= ****** GX       MC$NMR  001142        MD$CNA  000003        MD.UOB  000007        MF$LOO  177775
LOREOP= ****** GX       MC$NMS  001143        MD$COU  000001        MD.VOL  000006        MF$REA  000024
L$$ASG= 000000          MC$NNO  001606        MD$DA   000010        ME$ALI  000010        MF$SIG  177773
L$$DRV= 000000          MC$NNU  001605        MD$DL   000004        ME$AL2  000003        MF$SPF  000302
L$$P11= 000001          MC$NOP  001607        MD$DLV  000020        ME$ARE  000005        MF$SYS  000026
L$$11R= 000000          MC$NOR  000000        MD$DMC  000014        ME$CIR  000003        MF$TES  000022
MB$OIS  000001          MC$NPF  001616        MD$DMF  000046        ME$EXA  000000        MF$TRA  000301
```

```
PSIBYE     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              6-77       6-89       6-97
CALLX            #5-52       6-95
CEACC$           #5-52       6-89
PHBDF$           #5-52       5-54
RDTDF$           #5-52       5-55
RESMAP           #5-52       6-100
RETURN            6-101
SAVMAP           #5-52       6-78
SWSTK$            6-77
TCBDF$           #5-52       5-56
```

```
                                                                                         H 15
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04  Page 8-3
Symbol table

MP$SYP  002424          MS$FAI  000013          MT$CI   000007          MV$IV   000001          NM.INI  002400
MP$TFL  000156          MS$GLO  000000          MT$CLE  000077          MX$ACT  000020          NM.OPR  003000
MP$TH1  002206          MS$HOL  000002          MT$COD  000200          MX$CIR  000020          NM.VR2  002401
MP$TH2  002205          MS$INA  000002          MT$CON  000001          MX$CLN  000020          NM.VR3  002402
MP$TH3  002202          MS$LOA  000003          MT$DMC  000004          MX$CNM  000006          N$$ACC= 000001
MP$TLN  000202          MS$LOO  000002          MT$ETH  000006          MX$CON  000006          N$$BUF= 000001
MP$TLO  000172          MS$OFF  000001          MT$HEX  000040          MX$DAC  000020          N$$LDV= 000001
MP$TPA  004375          MS$ON   000000          MT$LPB  000005          MX$DTE  000020          N$$MCP= 000001
MP$TRI  002164          MS$PRO  000006          MT$MAX  000037          MX$FIL  000034          N$$MLL= 000001
MP$TST  000144          MS$REA  000004          MT$MUL  000100          MX$LGC  000777          N$$MDV= 000010
MP$TYP  002130          MS$REF  000001          MT$NLE  000017          MX$LIN  000020          N$$NCT= 000001
MP$UCS  004407          MS$RES  000003          MT$NON  000001          MX$LON  000006          N$$PEM= 000001
MP$USR  001750          MS$RST  000001          MT$NR4  000005          MX$NMS  000454          P$$P45= 000000
MP$VEC  004410          MS$RSX  000002          MT$NTY  000060          MX$NOD  000006          P$$WRD= 000000
MP$VER  004406          MS$RT   000005          MT$OCT  000060          MX$OBJ  000006          Q$$OPT= 000010
MP$WDF  002165          MS$SER  000002          MT$PHA  000002          MX$OWN  000040          R$$DER= 000000
MP$WMX  002177          MS$SHU  000002          MT$POI  000000          MX$PAR  000036          R$$K11= 000001
MP$XMT  002166          MS$SRV  000007          MT$QP2  000010          MX$PAS  000010          R$$SND= 000000
MP$XPF  000242          MS$STA  000000          MT$ROU  000000          MX$RAC  000047          R$$11M= 000000
MP$XPL  000244          MS$SYN  000012          MT$RO4  000004          MX$RID  000047          SAVNAM= ****** GX
MP$XPT  000240          MS$TER  000001          MT$SEC  000000          MX$RPS  000047          SF$ANM  000001
MP$XXX  177777          MS$TOP  000003          MT$SGD  000020          MX$SID  000040          SF$DNM  000002
MP$5ST  004406          MS$TRI  000005          MT$SYS  000002          MX$SNK  000377          S$$WRG= 000000
MS$ACT  000001          MS$UNR  000005          MT$TER  000001          MX$TYP  000077          S$$YSZ= 007600
MS$ADU  000010          MS$VMS  000004          MT$TRI  000002          MX$UID  000020          TSTALI= ****** GX
MS$ALO  000007          MS.DON  177600          MT$TYP  007777          M$$CRB= 000124          T$$KMG= 000000
MS$ASE  000006          MS.MOR  000002          MT$USD  000000          M$$CRX= 000000          T$$MIN= 000000
MS$ATR  000011          MS.PAR  000003          MT$X25  000003          M$$FCS= 000000          V$$CTR= 001000
MS$AUT  000000          MS.SUC  000001          MU$INC  000001          M$$MGE= 000000          X$$DBT= 000000
MS$CLE  000003          MT$AR4  000003          MU$OUT  000002          M$$NET= 000000          $SAVRG= ****** GX
MS$DED  000004          MT$ASC  000100          MU$PER  000000          M$$OVR= 000000          $$$SYS= 004374
MS$DIE  000003          MT$BIL  000001          MV$III  000000          NM.CLN  003400          $$$SYX= 000000
MS$DUM  000004          MT$BYS  000011

. ABS.  000103    000   (RW,I,GBL,ABS,OVR)
        000526    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 196
Work  file writes: 130
Size of work file: 25643 Words  ( 101 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:24.37
SY:ALCHIN.V2,[135,134]ALCHIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ALCHIN
```

```
                                                                          I 15
ALCHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:04      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
                                        .SBTTL  DEAANB - DEALLOCATE ALIAS NAME BLOCK
                                ;+
                                ; **-DEAANB-DEALLOCATE ALIAS NAME BLOCK
                                ;
                                ; UNLINKS THE BLOCK FROM THE ALIAS LIST, CALCULATES THE SIZE OF BLOCK,
                                ; AND DEALLOCATES IT.
                                ;
                                ; INPUTS:
                                ;       $PRUNM = UNMAPPED ADDRESS OF BLOCK BEFORE CURRENT BLOCK
                                ;       $CRUNM = UNMAPPED ADDRESS OF BLOCK TO BE DEQUEUED
                                ;       R1 = MAPPED ADDRESS OF BLOCK TO DEALLOCATE
                                ;
                                ; OUTPUTS:
                                ;       $PRUNM = UNMAPPED ADDRESS OF BLOCK BEFORE CURRENT BLOCK
                                ;       $CRUNM = UNMAPPED ADDRESS OF BLOCK BEFORE CURRENT BLOCK
                                ;       R1 = MAPPED ADDRESS OF PREVIOUS BLOCK IN LIST
                                ;
                                ; REGISTERS:
                                ;       R0 ,R3, R4, R5 ARE PRESERVED
                                ;-

000534                          DEAANB: SAVRG   <R0>                    ;; SAVE R0

                                        .IF DF  R$$PRO                  ;; PRO/DECNET

                                        MOV     @DECPT,R0               ;; POINT TO DECNET HOME BLOCK
                                        INC     D$ALNA(R0)              ;; INCREMENT # OF AVAILABLE ALIASES
                                        BIS     #DF.AFN,D$STAT(R0)      ;; SET UPDATE CONFIG. FILE BIT

                                        .ENDC

000536  010100                          MOV     R1,R0                   ;; COPY BLOCK POINTER
000540  062701  000020                  ADD     #A.ACC,R1               ;; POINT TO ID COUNT FIELD
000544  112102                          MOVB    (R1)+,R2                ;; GET LENGTH OF FIELD
000546  060201                          ADD     R2,R1                   ;; POINT TO PASSWORD FIELD
000550  112102                          MOVB    (R1)+,R2                ;; GET SIZE OF PASSWORD
000552  060201                          ADD     R2,R1                   ;; POINT TO ACCOUNT FIELD
000554  112102                          MOVB    (R1)+,R2                ;; GET SIZE OF ACCOUNT STRING
000556  060201                          ADD     R2,R1                   ;; POINT TO END OF BLOCK
000560  160001                          SUB     R0,R1                   ;; CALCULATE BLOCK LENGTH
000562  016700  000000G                 MOV     $CRUNM,R0               ;; GET THE UNMAPPED ADDRESS FOR DEACB
000566  016746  000000G                 MOV     $PRUNM,-(SP)            ;; SET UP FOR CALL TO $XLINK
000572  010046                          MOV     R0,-(SP)                ;; PUT PREVIOUS AND CURRENT PTRS ON STK
000574                                  CALLX   $XLINK,AUX              ;; LINK BLOCK INTO LIST
000604                                  CALL    DEACB                   ;; DEALLOCATE BLOCK
000610  016767  000000G 000000G         MOV     $PRUNM,$CRUNM           ;; NEW CURRENT ANB
000616  016746  000000G                 MOV     $CRUNM,-(SP)            ;; MAP TO PREVIOUS BLOCK
000622                                  CALL    @CEACC                  ;; ...
000626  012601                          MOV     (SP)+,R1                ;; GET MAPPED ADDRESS
000630                                  RESRG   <R0>                    ;; RESTORE R0
000632                                  RETURN
```

```
                .SBTTL  LOCFIL  - Locate Microcode File
        ;+
        ;                               *** LOCFIL ***
        ;
        ; Function:     To locate a microcode task image file on the current network
        ;               UIC of MLD's load device.  If the file is found, it is opened
        ;               for read with block I/O.
        ;
        ; Call Format:  CALL    LOCFIL
        ;
        ; Inputs:       R4 = address of filename/type/version ASCII string
        ;               R5 = length of the string
        ;
        ; Outputs:      C-bit clear = file found and opened
        ;               C-bit set   = file not found or could not be opened
        ;
        ;               BLOCK(+2) is set up for first file read
        ;
        ; Registers:    R0 through R5 destroyed
        ;
        ;-

        LOCFIL: CMP     R5,#MXFNLN              ; Does filename length exceed maximum?
                BLE     10$                     ; Branch if not
                MOV     #MXFNLN,R5              ; Else, force length to maximum

        10$:    MOV     R4,DSDS+10.             ; Move address of dataset descriptor
                MOV     R5,DSDS+8.              ; Move string length as well
                ADD     #4,DSDS+8.              ; Add in length of ".TSK" extension
                ADD     R5,R4                   ; Point past end of string
                MOVB    #'.,(R4)+               ; Append the string ".TSK"
                MOVB    #'T,(R4)+               ;   to the end of the specified
                MOVB    #'S,(R4)+               ;     filename string
                MOVB    #'K,(R4)                ; ...

                OPEN$R  #FDB,,,,,,FCSERR        ; Open the input file
                BCS     20$                     ; Branch if error occurred
                CLR     BLOCK                   ; Clear high virtual block number (VBN)
                MOV     #2,BLOCK+2              ; Set low VBN for first data block read

        20$:    RETURN                          ; Exit
```

```
                                                                  I  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31       PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

FR$UBU  = 000013            #6-83
FR$UPT  = 000014            #6-83
FS.HLT  = 000000             9-463        12-678
FS.RLB  = 002000             9-364
FS.WLB  = 001000             9-335         9-447        10-508
IO.WLB  = ******  GX         11-617
ISAR5     000570 RG         #8-229
I$$AS   = ******             6-84
KISAR5  = ******  GX         8-229
LDBADR    000150 R          #8-206        *9-334        10-528        10-532
LDBGT     000606 RG         #8-239         9-330
LDBIAS    000146 R          #8-205        *9-333        10-527
LDBPTR    000144 R          #8-204        *9-314        *9-332         9-338         9-371        *9-373         9-394        10-530        10-555
                             12-659
LDBRT     000610 RG         #8-240         9-372        12-661
LDBWTS  = 000001            #7-114         8-202
LF.ACT  = 100000            #6-85
LF.BRO  = 000400            #6-85
LF.BWT  = 000007            #6-85
LF.ENA  = 002000            #6-85
LF.LPB  = 001000            #6-85
LF.MDC  = 000100            #6-85
LF.MFL  = 004000            #6-85
LF.MTP  = 000020            #6-85
LF.PAC  = 000200            #6-85
LF.RDY  = 040000            #6-85
LF.REA  = 010000            #6-85
LF.SER  = 000040            #6-85
LF.TIM  = 000010            #6-85
LF.UNL  = 020000            #6-85
LF.X2P  = 000000            #6-85
LN.CLO  = 000000            #6-85
LN.DUM  = 000005            #6-85
LN.LOA  = 000004            #6-85
LN.LOO  = 000003            #6-85
LN.OAU  = 000003            #6-85
LN.OFF  = 000001            #6-85
LN.ON   = 000000            #6-85
LN.OOP  = 000004            #6-85
LN.OPE  = 000001            #6-85
LN.REF  = 000002            #6-85
LN.SER  = 000002            #6-85
LN.STA  = 000017            #6-85
LN.SUB  = 000360            #6-85
LN.TRI  = 000006            #6-85
LOCPAR    001074 R           9-438        #16-838
LOGEVT    000624 R           10-574       #11-599        16-876
L.COST    000015            #6-85
L.CTL     000012            #6-85          9-405
L.CVA     177776            #6-85
L.DDM     000002            #6-85          9-403
L.DDS     000004            #6-85          9-404
```

```
                                                                  J  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31       PAGE 5
SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
    215            000001               RF$PRV==1                          ; ... PRIVILEGED FUNCTION
    216            000002               RF$NCX==2                          ; ... FUNCTION DOES NOT REQUIRE A CONTEXT
    217 000003                                  .BLKB   1                  ; <UNUSED>
    218 000004                          R$INIT::.BLKW   1                  ; ADDRESS OF INITIALIZE ROUTINE
    219 000006                          R$OPER::.BLKW   1                  ; ADDRESS OF OPERATE ROUTINE
    220 000010                          R$SIZE::                           ; LENGTH OF BLOCK
    221 000204                                  .PSECT
    222
    223                                 ;
    224                                 ; Saved register offsets for SWSTK$
    225                                 ;
    226            000002                       R$R0 == 2                  ; SAVED R0
    227            000004                       R$R1 == 4                  ; SAVED R1
    228            000006                       R$R2 == 6                  ; SAVED R2
    229            000010                       R$R3 == 10                 ; SAVED R3
    230            000012                       R$R4 == 12                 ; SAVED R4
    231            000014                       R$R5 == 14                 ; SAVED R5
    232            000016                       R$PC == 16                 ; SAVED PC
    233
    234                                 ;
    235                                 ; BIT POSITION DEFINITIONS
    236                                 ;
    237            000001                       BIT0 == 1
    238            000002                       BIT1 == 2
    239            000004                       BIT2 == 4
    240            000010                       BIT3 == 10
    241            000020                       BIT4 == 20
    242            000040                       BIT5 == 40
    243            000100                       BIT6 == 100
    244            000200                       BIT7 == 200
    245            000400                       BIT8 == 400
    246            001000                       BIT9 == 1000
    247            002000                       BIT10 == 2000
    248            004000                       BIT11 == 4000
    249            010000                       BIT12 == 10000
    250            020000                       BIT13 == 20000
    251            040000                       BIT14 == 40000
    252            100000                       BIT15 == 100000
    253
    254
    255                                 ;
    256                                 ; STORAGE
    257                                 ;
    258 000204                          $PRUNM::.BLKW   1                  ; UNMAPPED ADDRESS OF PREV XPOOL BLOCK
    259 000206                          $CRUNM::.BLKW   1                  ; MAPPED ADDRESS OF PREV XPOOL BLOCK
    260 000210                          CURIOP::.BLKW   1                  ; ADDRESS OF CURRENT I/O PACKET
    261 000212                          CURCTX::.BLKW   1                  ; ADDRESS OF CURRENT CONTEXT AREA
    262 000214     000000               LSTCTX::.WORD   0                  ; ADDRESS OF LAST CONTEXT AREA USED
    263 000216                          DELFLG: .BLKW   1                  ; CURRENT CONTEXT SHOULD BE DELETED
    264 000220     000000               NUMUSR::.WORD   0                  ; NUMBER OF CURRENT USERS
    265 000222                          TSKLN2::.BLKW   1                  ; COPY OF TASK'S 2ND LUN WORD
    266 000224     000000               FDNCNT::.WORD   0                  ; COUNT OF FORKS "DONE"
    267 000226                          ISTACK::.BLKW   1                  ; INITIAL TASK SP
    268 000230                          STATUS::.BLKW   1                  ; WORD FOR RETURNING FORK STATUS
    269 000232     000000               CTXHD:: .WORD   0                  ; CONTEXT BLOCK LISTHEAD
    270 000234     000000               POOLPT::.WORD   0                  ; FIRST FREE BYTE IN CONTEXT POOL
    271 000236                          POOLSZ::.BLKW   1                  ; NUMBER OF FREE BYTES AT POOLPT
```

```
FS.REA= 001000          IE.CON= 177752          IE.RER= 177740          I.ATTL= 000044          MAXLID= 000020
FS.RET= 000000          IE.DAA= 177770          IE.RES= 177644          I.EFN   000003 G        MB$DIS  000001
FS.REZ= 003000          IE.DAO= 177763          IE.RNM= 177715          I.FCN   000012 G        MB$ENA  000000
FS.RLB= 002000          IE.DFU= 177750          IE.RSU= 177757          I.IOSB  000014 G        MB$FUL  000002
FS.RNG= 011000          IE.DIS= 177673          IE.SNC= 177735          I.LGTH= 000044          MB$MIX  000002
FS.RST= 000000          IE.DNA= 177771          IE.SPC= 177772          I.LNK   000000 G        MB$ONE  000001
FS.RTN= 001000          IE.DNR= 177775          IE.SPI= 177634          I.LN2   000006 G        MB$RXO  000001
FS.SET= 005000          IE.DSQ= 177646          IE.SQC= 177734          I.PRI   000002 G        MB$TXO  000000
FS.SFC= 005000          IE.DUN= 177767          IE.SRE= 177762          I.PRM   000024 G        MB$ZER  000000
FS.SFR= 006000          IE.DUP= 177707          IE.STK= 177706          I.TCB   000004 G        MC$BAB  006073
FS.SFS= 004000          IE.EOF= 177766          IE.SZE= 177636          I.UCB   000010 G        MC$BID  001765
FS.SPW= 040000          IE.EOT= 177702          IE.TML= 177643          I.XDBF  000040 G        MC$BMC  001767
FS.STM= 000000          IE.EOV= 177765          IE.TMO= 177641          I.XDBL  000044 G        MC$BSC  001766
FS.STP= 002000          IE.EXP= 177665          IE.UKN= 177637          I.XIOP  000002 G        MC$CAC  001442
FS.STR= 001000          IE.FEX= 177717          IE.ULK= 177653          I.XLEN= 000046          MC$CAP  001440
FS.TRM= 003000          IE.FHE= 177705          IE.URJ= 177667          I.XLNK  000000 G        MC$CCL  001445
FS.WLB= 001000          IE.FLG= 177647          IE.VER= 177774          I.XMOD  000006 G        MC$CDC  002045
FS.XKL= 002000          IE.FLN= 177657          IE.WAC= 177743          I.XPBF  000022 G        MC$CDP  001441
FS.XOF= 010000          IE.FOP= 177713          IE.WAT= 177741          I.XPBL  000026 G        MC$CIF  001465
FS.XON= 007000          IE.HFU= 177744          IE.WCK= 177652          I.XPBV  000030 G        MC$CLD  001464
FS.ZER= 002000          IE.ICE= 177721          IE.WER= 177737          I.XRBF  000012 G        MC$COU  100000
FTABLE= ****** GX       IE.IES= 177656          IE.WLK= 177764          I.XRBL  000016 G        MC$CTL  001454
F$$LVL= 000001          IE.IFC= 177776          IE.2DV= 177720          I.XTCB  000004 G        MC$CTR  001452
GETPKT  001502RG        IE.IFU= 177747          IFCERR  002226R         I.XTMO  000020 G        MC$CTS  001453
GS.DEL= 000001          IE.ILL= 177726          INIACP= ****** GX       I.XTTB  000032 G        MC$DOV  002050
G$$TPP= 000000          IE.IQU= 177645          INIREQ= ****** GX       I.XTTL  000036 G        MC$LBR  001750
G$$TSS= 000000          IE.ISQ= 177703          IOFIN   000032RG        KISAR5= ****** GX       MC$LBS  001751
G$$TTK= 000000          IE.LCK= 177745          IP.ABO= 000040          KISAR6= ****** GX       MC$LDI  001774
G$$WRD= 000000          IE.MII= 177635          IP.FAK= 000020          KSAR6   000006RG        MC$LDO  001775
G.CNT   000004 G        IE.MOD= 177753          IP.PND= 000100          K$$CNT= 177546          MC$LDR  001762
G.EFLG  000006 G        IE.NBF= 177731          IP.UMR= 000200          K$$CSR= 177546          MC$LDS  001763
G.GRP   000002 G        IE.NBK= 177727          ISAR5   000002RG        K$$LDC= 000000          MC$LLB  002021
G.LGTH= 000012          IE.NDA= 177662          ISAR6   000004RG        K$$TPS= 000074          MC$LLP  002115
G.LNK   000000 G        IE.NDR= 177670          ISTACK  000226RG        LC$EXT  000000          MC$LLR  002007
G.STAT  000003 G        IE.NFI= 177704          IS.BV = 000005          LC$INT  000001          MC$LOO  000001
HEADR   000030RG        IE.NFW= 177673          IS.CC = 001401          LDBAF   000074RG        MC$LRB  002020
IDBUFF  001410RG        IE.NLK= 177661          IS.CHW= 000004          LDBGT   000076RG        MC$LRP  002114
IE.ABO= 177761          IE.NLN= 177733          IS.CNV= 000003          LDBRT   000100RG        MC$LRR  002006
IE.ALC= 177654          IE.NNC= 177663          IS.CR = 006401          LD$LP = 000000          MC$LSI  002032
IE.ALN= 177736          IE.NNN= 177674          IS.DAO= 000002          LGCON   000102RG        MC$LST  002033
IE.BAD= 177777          IE.NNT= 177642          IS.EOT= 002001          LGFNB   000104RG        MC$LUP  006072
IE.BBE= 177710          IE.NOD= 177751          IS.ESC= 015401          LGMON   000106RG        MC$MAP  010000
IE.BCC= 177676          IE.NRJ= 177666          IS.ESQ= 115401          LGSTT   000110RG        MC$MBL  001764
IE.BDI= 177714          IE.NSF= 177746          IS.OOB= 000003          LGUIC   000112RG        MC$MBX  004420
IE.BDR= 177716          IE.NST= 177660          IS.PES= 100001          LLCRS   000114RG        MC$MBY  001752
IE.BDV= 177711          IE.NTR= 177651          IS.PND= 000000          LLCTA   000116RG        MC$MPX  004406
IE.BHD= 177700          IE.OFL= 177677          IS.RDD= 000002          LSTCTX  000214RG        MC$NAP  001604
IE.BLB= 177672          IE.ONL= 177675          IS.RVI= 000002          L$$ASG= 000000          MC$NBR  001130
IE.BLK= 177754          IE.ONP= 177773          IS.SUC= 000001          L$$DRV= 000000          MC$NBS  001131
IE.BNM= 177712          IE.OVR= 177756          IS.TAB= 004401          L$$P11= 000001          MC$NCR  001154
IE.BTF= 177664          IE.PES= 177655          IS.TMM= 000004          L$$11R= 000000          MC$NCS  001155
IE.BTP= 177725          IE.PRI= 177760          IS.TMO= 000002          MANACP  001472RG        MC$NML  001274
IE.BVR= 177701          IE.RAC= 177724          IS.TNC= 000002          MANCLN  002702R         MC$NMR  001142
IE.BYT= 177755          IE.RAT= 177723          IS.XPT= 000005          MANINI  002300R         MC$NMS  001143
IE.CKS= 177742          IE.RBG= 177730          I$$RAR= 000000          MANOPR  002626R         MC$NNO  001606
IE.CLO= 177732          IE.RCN= 177722          I$$RDN= 000000          MANTBL  000240R         MC$NNU  001605
IE.CNR= 177640          IE.REJ= 177650          I.AST   000022 G        MAXID = 000062          MC$NOP  001607
```

```
		.TITLE	MANAST - NETWORK MANAGEMENT ACP AST PROCESSING
		.IDENT	/V05.00/
;
; COPYRIGHT (C) 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;	AST PROCESSING ROUTINES FOR THE NETWORK MANAGEMENT ACP
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 4.01	17-OCT-84			RJK
;	Change Symbols for Vectoring Comm Exec
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;
;
```

```
    116 000174  010377  000000G                 MOV     R3,@NMCL2               ;; ELSE, FIX UP LIST TAIL POINTER; RJK02
    117 000200  116400  000003          22$:    MOVB    C.BID(R4),R0            ;; GET BUFFER ID FIELD
    118 000204  042700  177760                  BIC     #^C<17>,R0              ;; ISOLATE CODE
    119 000210  016000  000304'                 MOV     BUFTBL-2(R0),R0         ;; RELEASE THE BUFFER
    120 000214                                  CALL    @(R0)                   ;; ...
    121 000220  000742                          BR      20$                     ;; ... AND LOOP
    122 000222                          25$:    .ENDC   ;DF MN$VOL
    123
    124 000222                                  RETURN                          ;; BACK TO USER STATE
    125
    126 000224  012767  000000' 000000G 30$:    MOV     #CTXPOO,POOLPT          ; SET ADDRESS OF FREE POOL
    127 000232                                  GTSK$S  #CTXPOO                 ; GET TASK PARAMETERS
    128 000244  103417                          BCS     100$                    ; IF CS, RETURN WITH ERROR
    129 000246  010601                          MOV     SP,R1                   ; GET STACK POINTER
    130 000250  042701  017777                  BIC     #17777,R1               ; CLEAR ALL BUT APR BIAS
    131 000254  066701  177552                  ADD     CTXPOO+G.TSTS,R1        ; ADD TASK SIZE TO CALC HI VIRT ADDRESS
    132 000260  162701  000002'                 SUB     #CTXPOO+2,R1            ; CALCULATE CONTEXT POOL SIZE
    133 000264  010167  000000G                 MOV     R1,POOLSZ               ; ... AND SAVE IT
    134 000270                                  SREA$S  #REXAST                 ; SPECIFY REQUESTED EXIT AST
    135 000302  000241                          CLC                             ; ... IGNORE ERROR
    136 000304                          100$:   RETURN
    137
    138                                 ;
    139                                 ; SYSTEM BUFFER RETURN DISPATCH TABLE
    140                                 ; THIS TABLE IS USED FOR RELEASING BUFFERS LEFT FROM SYSTEM LEVEL COMPLETIONS
    141                                 ;
    142                                         .IF DF  MN$VOL                  ; USED IN VOLATILE ACP ONLY
    143
    144 000306  000000G                 BUFTBL: .WORD   CCBRT                   ; RETURN A CCB
    145 000310  000000G                         .WORD   LDBRT                   ; RETURN A LARGE DATA BUFFER
    146 000312  000313'                         .WORD   .+1                     ; RESERVED
    147 000314  000000G                         .WORD   CSBRT                   ; RETURN A COMBINED CCB AND SMALL DATA BUFFER
    148
    149                                         .ENDC   ;DF MN$VOL
    150
    151 000000                                  .PSECT
```

I 7

Table of contents

```
     5-   52     MACRO CALLS AND LOCAL DATA
     6-   86     LOGOUT - PERFORM LOGOUT CLEANUP FOR A USER
     7-  119     DECBYE - LOGOUT CLEANUP FOR DECNET
```

J 7

```
**FILE**ID**MANFRK
```

```
MM      MM    AAAAAA    NN      NN  FFFFFFFFFF  RRRRRRRR    KK      KK
MM      MM    AAAAAA    NN      NN  FFFFFFFFFF  RRRRRRRR    KK      KK
MMMM  MMMM  AA      AA  NN      NN  FF          RR      RR  KK      KK
MMMM  MMMM  AA      AA  NN      NN  FF          RR      RR  KK      KK
MM  MM  MM  AA      AA  NNNN    NN  FF          RR      RR  KK    KK
MM  MM  MM  AA      AA  NNNN    NN  FF          RR      RR  KK    KK
MM      MM  AA      AA  NN  NN  NN  FFFFFFFF    RRRRRRRR    KKKKKK
MM      MM  AA      AA  NN  NN  NN  FFFFFFFF    RRRRRRRR    KKKKKK
MM      MM  AAAAAAAAAA  NN    NNNN  FF          RR  RR      KK    KK
MM      MM  AAAAAAAAAA  NN    NNNN  FF          RR  RR      KK    KK
MM      MM  AA      AA  NN      NN  FF          RR    RR    KK      KK     ....
MM      MM  AA      AA  NN      NN  FF          RR    RR    KK      KK     ....
MM      MM  AA      AA  NN      NN  FF          RR      RR  KK      KK     ....
MM      MM  AA      AA  NN      NN  FF          RR      RR  KK      KK     ....
```

```
LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
MANFRK - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:30
Table of contents
```

I 9

Table of contents

```
     5-   51     MACRO CALLS AND LOCAL DATA
     6-  126     INIQIO - ISSUE NM.INI QIO TO NM:
     6-  127     OPRQIO - ISSUE NM.OPR QIO TO NM:
     7-  254     PRIVILEGED ROUTINE ENTRY POINTS
```

J 9

```
MP$SDV   000160       MP$XPL   000244       MT$ASC   000100       MX$FIL   000034       OBCHIN   000640RG
MP$SEH   004401       MP$XPT   000240       MT$BIL   000001       MX$LGC   000777       OBCHOP   000722RG
MP$SER   000144       MP$XXX   177777       MT$BYS   000011       MX$LIN   000020       OBID     000652R
MP$SET   000000       MP$5ST   004406       MT$CI    000007       MX$LON   000006       OBREIN   000646RG
MP$SGZ   001644       MS$ACT   000001       MT$CLE   000077       MX$NMS   000454       OBREOP   000722RG
MP$SID   000176       MS$ADU   000010       MT$COD   000200       MX$NOD   000006       OPRQIO   000156R
MP$SIN   000310       MS$ALO   000007       MT$CON   000001       MX$OBJ   000006       PRCHIN   000506RG
MP$SLI   000156       MS$ASE   000006       MT$DMC   000004       MX$OWN   000040       PRCHOP   000722RG
MP$SLO   000171       MS$ATR   000011       MT$ETH   000006       MX$PAR   000036       PRID     000520R
MP$SMX   002202       MS$AUT   000000       MT$HEX   000040       MX$PAS   000010       PRREIN   000514RG
MP$SND   000500       MS$CLE   000003       MT$LPB   000005       MX$RAC   000047       PRREOP   000722RG
MP$SNP   006332       MS$DED   000004       MT$MAX   000037       MX$RID   000047       P$$P45=  000000
MP$SNU   000542       MS$DIE   000003       MT$MUL   000100       MX$RPS   000047       P$$WRD=  000000
MP$SOB   000524       MS$DUM   000004       MT$NLE   000017       MX$SID   000040       Q$$OPT=  000010
MP$SPA   000157       MS$FAI   000013       MT$NON   000001       MX$SNK   000377       R$$DER=  000000
MP$SPR   000536       MS$GLO   000000       MT$NR4   000005       MX$TYP   000077       R$$K11=  000001
MP$SPS   000513       MS$HOL   000002       MT$NTY   000060       MX$UID   000020       R$$SND=  000000
MP$STA   000000       MS$INA   000002       MT$OCT   000060       M$$CRB=  000124       R$$11M=  000000
MP$STI   002140       MS$LOA   000003       MT$PHA   000002       M$$CRX=  000000       SF$ANM   000001
MP$STT   002201       MS$LOO   000002       MT$POI   000000       M$$FCS=  000000       SF$DNM   000002
MP$STY   000175       MS$OFF   000001       MT$QP2   000010       M$$MGE=  000000       SYCHIN   000666RG
MP$SUB   000001       MS$ON    000000       MT$ROU   000000       M$$NET=  000000       SYCHOP   000722RG
MP$SUR   000156       MS$PRO   000006       MT$RO4   000004       M$$OVR=  000000       SYID     000706R
MP$SUS   000512       MS$REA   000004       MT$SEC   000000       NMSTAT   000010R      SYREIN   000674RG
MP$SVR   000163       MS$REF   000001       MT$SGD   000020       NM.CLN   003400       SYREOP   000722RG
MP$SWI   002260       MS$RES   000003       MT$SYS   000002       NM.INI   002400       SYZEIN   000702RG
MP$SYP   002424       MS$RST   000001       MT$TER   000001       NM.OPR   003000       SYZEOP   000722RG
MP$TFL   000156       MS$RSX   000002       MT$TRI   000002       NM.VR2   002401       S$$WRG=  000000
MP$TH1   002206       MS$RT    000005       MT$TYP   007777       NM.VR3   002402       S$$YSZ=  007600
MP$TH2   002205       MS$SER   000002       MT$USD   000000       NOCHIN   000600RG     T$$KMG=  000000
MP$TH3   002202       MS$SHU   000002       MT$X25   000003       NOCHOP   000722RG     T$$MIN=  000000
MP$TLN   000202       MS$SRV   000007       MU$INC   000001       NOID     000620R      V$$CTR=  001000
MP$TLO   000172       MS$STA   000000       MU$OUT   000002       NOREIN   000606RG     X$$DBT=  000000
MP$TPA   004375       MS$SYN   000012       MU$PER   000000       NOREOP   000722RG     $CBTA =  ****** GX
MP$TRI   002164       MS$TER   000001       MV$III   000000       NOZEIN   000614RG     $DSW  =  ****** GX
MP$TST   000144       MS$TOP   000003       MV$IV    000001       NOZEOP   000722RG     $TMEFN=  ****** GX
MP$TYP   002130       MS$TRI   000005       MXMES =  000454       N$$ACC=  000001       $TMLUN=  ****** GX
MP$UCS   004407       MS$UNR   000005       MX$ACT   000020       N$$BUF=  000001       $$$   =  000235R      002
MP$USR   001750       MS$VMS   000004       MX$CIR   000020       N$$LDV=  000001       $$$ARG=  000002
MP$VEC   004410       MS.DON   177600       MX$CLN   000020       N$$MCP=  000001       $$$OST=  000010
MP$VER   004406       MS.MOR   000002       MX$CNM   000006       N$$MLL=  000001       $$$SYS=  004374
MP$WDF   002165       MS.PAR   000003       MX$CON   000006       N$$MOV=  000010       $$$SYX=  000000
MP$WMX   002177       MS.SUC   000001       MX$DAC   000020       N$$NCT=  000001       $$$T1 =  000002
MP$XMT   002166       MT$AR4   000003       MX$DTE   000020       N$$PEM=  000001       $$$T2 =  000004R
MP$XPF   000242

. ABS.   000000     000   (RW,I,GBL,ABS,OVR)
         000726     001   (RW,I,LCL,REL,CON)
$$$STR   000272     002   (RO,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 200
Work  file writes: 130
Size of work file: 25610 Words  ( 101 Pages)
```

```
                                         .SBTTL  TRAACC - TRANSLATE AN ALIAS TO AN NAME AND ACCESS CONTROL
                                 ;+
                                 ; **-TRAACC-TRANSLATE AN ALIAS TO A NAME AND ACCESS CONTROL FIELDS
                                 ;
                                 ; THIS ROUTINE TRANSLATES AN ALIAS TO A NAME AND ACCESS CONTROL FIELDS
                                 ;
                                 ; INPUTS:
                                 ;       R1 = POINTER TO ALIAS
                                 ;       R3 = ADDRESS OF BUFFER
                                 ;       CURIOP = ADDRESS OF CURRENT I/O PACKET
                                 ;
                                 ; OUTPUTS:
                                 ;       R0 = RETURN STATUS
                                 ;       R2 = LENGTH OF DATA TO RETURN
                                 ;-

000216                           TRAACC::
000216  005002                           CLR     R2                      ; ASSUME NO NAME TO RETURN
000220  016705  000000G                  MOV     CURIOP,R5               ; GET ADDRESS OF I/O PACKET
000224  026527  000006G 000006           CMP     I.PRM+6(R5),#6          ; WAS A SIX BYTE ALIAS SUPPLIED ?
000232  001403                           BEQ     10$                     ; IF EQ, YES - TRY TO MAP IT
000234  012700  177776                   MOV     #ME.FOR,R0              ; ELSE, MESSAGE FORMAT ERROR
000240  000462                           BR      90$                     ; AND RETURN

000242  010100                   10$:    MOV     R1,R0                   ; POINT TO THE ALIAS
000244                                   SWSTK$  50$                     ;; ENTER SYSTEM STATE
000250                                   SAVMAP                          ;; SAVE CURRENT MAPPING
000254                                   CALL    REPALI                  ;; TRY TO FIND AND REPLACE ALIAS
000260  103006                           BCC     20$                     ;; IF CC, GOT THE NAME
000262                                   RETC    R0                      ;; ELSE, RETURN WITH ERROR
000274  000421                           BR      40$                     ;; ...
000276                           20$:
000276  062703  000020                   ADD     #A.ACC,R3               ;; POINT TO ACCESS CONTROL
000302  010301                           MOV     R3,R1                   ;; ...
000304  112102                           MOVB    (R1)+,R2                ;; GET LENGTH OF USER FIELD
000306  060201                           ADD     R2,R1                   ;; POINT TO NEXT FIELD
000310  112102                           MOVB    (R1)+,R2                ;; GET LENGTH OF PASSWORD FIELD
000312  060201                           ADD     R2,R1                   ;; POINT TO NEXT FIELD
000314  112102                           MOVB    (R1)+,R2                ;; GET LENGTH OF ACCOUNT FIELD
000316  060201                           ADD     R2,R1                   ;; POINT TO END OF ACCESS CONTROL INFO
000320  160301                           SUB     R3,R1                   ;; CALCULATE NUMBER OF BYTES TO COPY
000322  010166  000006                   MOV     R1,4+BIAS(SP)           ;; RETURN R1 IN USER R1
000326  005701                           TST     R1                      ;; ANY ACCESS CONTROL TO COPY?
000330  001403                           BEQ     40$                     ;; BR IF NO
000332  112320                   30$:    MOVB    (R3)+,(R0)+             ;; STORE ACCESS CONTROL INFO
000334  005301                           DEC     R1                      ;; MORE TO COPY?
000336  003375                           BGT     30$                     ;; BR IF YES
000340                           40$:    RESMAP                          ;; RESTORE PREVIOUS MAPPING
000344                                   RETURN                          ;; BACK TO USER STATE AND RETURN

000346  103003                   50$:    BCC     60$                     ; IF CC, SUCCESS
000350  012700  177770                   MOV     #ME.UCO,R0              ; ELSE "UNKNOWN COMPONENT"
000354  000414                           BR      90$                     ; ...
000356  010302                   60$:    MOV     R3,R2                   ; SAVE START OF OUTPUT BUFFER
000360  012022                           MOV     (R0)+,(R2)+             ; SET THE NAME IN THE BUFFER
000362  012022                           MOV     (R0)+,(R2)+             ; ...
000364  012022                           MOV     (R0)+,(R2)+             ; ...
```

```
                                                   I 12
MANVCD - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:31  Page 7-1
Symbol table

A$$CHK= 000000          MC$CTL  001454          MC$UBU  002052          MD$KMX  000054          ME.FUN  177777
A$$CPS= 000000          MC$CTR  001452          MC$UFD  002047          MD$KMY  000052          ME.GRO  177745
A$$PRI= 000000          MC$CTS  001453          MC$UMR  001132          MD$LEN  000050          ME.HAR  177750
A$$TRP= 000000          MC$DOV  002050          MC$UMS  001133          MD$PCL  000011          ME.IID  177767
CM$RSX= 000200          MC$LBR  001750          MC$VAX  000003          MD$QNA  000005          ME.LCO  177766
C$$CKP= 000000          MC$LBS  001751          MC$WID  060000          MD$UNA  000001          ME.LPR  177757
C$$ORE= 000400          MC$LDI  001774          MC$WIH  040000          MD$WIT  000002          ME.MPR  177773
C$$RSH= 177564          MC$LDO  001775          MC$WIL  020000          MD.ABO  000017          ME.MVE  177771
D$$BUG= 177514          MC$LDR  001762          MC$W08  020000          MD.ACC  000010          ME.OPE  177747
D$$ISK= 000000          MC$LDS  001763          MC$W16  040000          MD.AOB  000016          ME.PLO  177751
D$$L11= 000001          MC$LLB  002021          MC$W32  060000          MD.BOB  000011          ME.PMI  177743
D$$YNC= 000000          MC$LLP  002115          MC$XBR  001750          MD.DIA  000007          ME.PNA  177752
D$$YNM= 000000          MC$LLR  002007          MC$XBS  001751          MD.DOB  000015          ME.PRI  177775
E$$XPR= 000000          MC$LOO  000001          MC$XCJ  000322          MD.DUM  000002          ME.PTY  177772
FE.CEX= 020000          MC$LRB  002020          MC$XCR  002260          MD.FAI  000014          ME.PVA  177760
FMASK = ****** GX       MC$LRP  002114          MC$XCS  002261          MD.FNA  000001          ME.RES  177761
F$$LVL= 000001          MC$LRR  002006          MC$XDR  001762          MD.FOB  000006          ME.ROO  177754
G$$TPP= 000000          MC$LSI  002032          MC$XDS  001763          MD.LOA  000001          ME.SIZ  177774
G$$TSS= 000000          MC$LST  002033          MC$XFR  002272          MD.LSH  000020          ME.SYS  177746
G$$TTK= 000000          MC$LUP  006072          MC$XFS  002273          MD.NOB  000012          ME.UCO  177770
G$$WRD= 000000          MC$MAP  010000          MC$XLJ  000323          MD.NON  177777          MF$ACT  177776
INIACP  000000RG        MC$MBL  001764          MC$XLR  002330          MD.PER  000000          MF$ADD  000000
INIBUF= ****** GX       MC$MBX  004420          MC$XMA  000310          MD.RES  000004          MF$ADJ  177774
INIREQ  000014RG        MC$MBY  001752          MC$XMC  002305          MD.ROB  000005          MF$ALL  177775
I$$RAR= 000000          MC$MPX  004406          MC$XMS  002304          MD.RSH  000013          MF$BYE  000300
I$$RDN= 000000          MC$NAP  001604          MC$XNR  002332          MD.SDU  000005          MF$CHA  000023
K$$CNT= 177546          MC$NBR  001130          MC$XRC  002316          MD.SLO  000003          MF$DUM  000020
K$$CSR= 177546          MC$NBS  001131          MC$XRR  002331          MD.SNA  000000          MF$EVT  000001
K$$LDC= 000000          MC$NCR  001154          MC$XRS  002342          MD.TLO  000004          MF$FUP  000264
K$$TPS= 000074          MC$NCS  001155          MC$020  000002          MD.UNA  000002          MF$KNO  177777
LC$EXT  000000          MC$NML  001274          MD$CI   000007          MD.UNR  000003          MF$LOA  000017
LC$INT  000001          MC$NMR  001142          MD$CNA  000003          MD.UOB  000007          MF$LOO  177775
LD$LP = 000000          MC$NMS  001143          MD$COU  000001          MD.VOL  000006          MF$REA  000024
L$$ASG= 000000          MC$NNO  001606          MD$DA   000010          ME$ALI  000010          MF$SIG  177773
L$$DRV= 000000          MC$NNU  001605          MD$DL   000004          ME$AL2  000003          MF$SPF  000302
L$$P11= 000001          MC$NOP  001607          MD$DLV  000020          ME$ARE  000005          MF$SYS  000026
L$$11R= 000000          MC$NOR  000000          MD$DMC  000014          ME$CIR  000003          MF$TES  000022
MB$DIS  000001          MC$NPF  001616          MD$DMF  000046          ME$EXA  000000          MF$TRA  000301
MB$ENA  000000          MC$NPR  001630          MD$DMP  000022          ME$EXE  000200          MF$TRI  000021
MB$FUL  000002          MC$NRE  001200          MD$DMR  000050          ME$LIN  000001          MF$ZER  000025
MB$MIX  000002          MC$NRT  001166          MD$DMV  000042          ME$LOG  000002          ML$ALL  100000
MB$ONE  000001          MC$NVR  001642          MD$DN   000016          ME$MOD  000004          ML$CLS  000000
MB$RXO  000001          MC$P08  000000          MD$DP   000000          ME$NOD  000000          ML$CON  000001
MB$TXO  000000          MC$P11  000001          MD$DPV  000044          ME$NON  177777          ML$EXT  000001
MB$ZER  000000          MC$RFL  002046          MD$DQ   000006          ME$OBJ  000007          ML$FIL  000002
MC$BAB  006073          MC$SBU  002051          MD$DTE  000024          ME$OB2  000004          ML$FIR  000000
MC$BID  001765          MC$SFL  002044          MD$DU   000002          ME$PRO  000005          ML$INT  000000
MC$BMC  001767          MC$SYC  000012          MD$DUP  000012          ME$SYS  000006          ML$KNO  140000
MC$BSC  001766          MC$SYL  000036          MD$DV   000026          ME.BLO  177744          ML$MON  000003
MC$CAC  001442          MC$SYR  000050          MD$DZ   000030          ME.CON  177753          ML$SYS  000002
MC$CAP  001440          MC$SYS  000024          MD$FUL  000000          ME.CST  177765          ML$TOP  000001
MC$CCL  001445          MC$SYZ  000000          MD$HAL  000001          ME.DIS  177755          MN$UNL  000377
MC$CDC  002045          MC$TBR  001140          MD$HEL  000002          ME.DON  177600          MO$ACC  000200
MC$CDP  001441          MC$TBS  001141          MD$KCP  000013          ME.FCO  177762          MO$ADD  000002
MC$CIF  001465          MC$TDR  002053          MD$KDP  000034          ME.FIO  177756          MO$ALA  000004
MC$CLD  001464          MC$TIM  000000          MD$KDZ  000036          ME.FOP  177763          MO$ALI  000000
MC$COU  100000          MC$TYP  007777          MD$KL   000040          ME.FOR  177776          MO$CHA  000040
```

```
                                                   J 12
MANVCD - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:31  Page 7-2
Symbol table
```

```
                                                    I 13
MANVTB - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:32  Page 6-4
Symbol table

MF$TRI   000021            MP$ADJ   001440            MP$DIA   000173            MP$MAC   004420            MP$PAR   000036
MF$ZER   000025            MP$ADP   006325            MP$DLB   002571            MP$MAD   001630            MP$PAS   000513
ML$ALL   100000            MP$ADS   004406            MP$DLG   004521            MP$MAP   006323            MP$PCH   002152
ML$CLS   000000            MP$ALB   000036            MP$DLI=  001466            MP$MAR   001635            MP$PCO   000024
ML$CON   000001            MP$AMC   001640            MP$DLT   002200            MP$MAV   006321            MP$PCT   000144
ML$EXT   000001            MP$AMH   001641            MP$DRO   001441            MP$MBN   001636            MP$PDT   002114
ML$FIL   000002            MP$ANB   004420            MP$DST   000454            MP$MBR   001637            MP$PHA   000170
ML$FIR   000000            MP$ASB   000024            MP$DTE   002140            MP$MBU   001642            MP$PHY   000012
ML$INT   000000            MP$ASC   004432            MP$DTY=  001452            MP$MCB   000156            MP$PLI   000025
ML$KNO   140000            MP$AUS   000512            MP$DUA   000207            MP$MCO   001632            MP$PLN   002140
ML$MON   000003            MP$BBT   002165            MP$DUC   000210            MP$MDE   004421            MP$PLO   000012
ML$SYS   000002            MP$BDF   002164            MP$DUM   000202            MP$MHO   001633            MP$PMC   004374
ML$TOP   000001            MP$BFQ   002121            MP$DUP   002127            MP$MLB   000202            MP$PNT   002126
MN$UNL   000377            MP$BLK   001616            MP$DVC   002114            MP$MLK   001306            MP$PRI   004411
MOCHIN=  ******  GX        MP$BLO   001452            MP$DWE   001321            MP$MLN   001631            MP$PRO   002130
MOCHOP=  ******  GX        MP$BMX   002176            MP$ELT   000157            MP$MLP   006333            MP$PSS   001763
MOREIN=  ******  GX        MP$BNP   006330            MP$ETY   001605            MP$MRB   002171            MP$PST   001762
MOREOP=  ******  GX        MP$BRT   001620            MP$EVE   000311            MP$MRO   001605            MP$RET   001631
MOZEIN=  ******  GX        MP$BSA   002176            MP$FNC   001752            MP$MRP   006322            MP$RFA   001323
MO$ACC   000200            MP$BSD   002203            MP$GDT   002222            MP$MRT   002153            MP$RMX   002201
MO$ADD   000002            MP$BSI   002200            MP$GNM   002223            MP$MRV   006320            MP$RPA   004374
MO$ALA   000004            MP$BSP   006331            MP$GRO   002115            MP$MSB   000170            MP$RPR   001606
MO$ALI   000000            MP$BUF   000170            MP$GRP   000541            MP$MVE   000145            MP$RRT   002213
MO$CHA   000040            MP$BUP   006324            MP$GTY   002224            MP$MVI   001634            MP$RST   002212
MO$CIR   000003            MP$BUS   001643            MP$HAD   001757            MP$MVR   001751            MP$RSV   001754
MO$CLE   000100            MP$CAC   001750            MP$HBT   002142            MP$MWN   002154            MP$RSZ   001756
MO$COU   000060            MP$CAS   001762            MP$HDD   000162            MP$MXB   002172            MP$RTI   001616
MO$DAC   000000            MP$CAT   002210            MP$HTM   001612            MP$MXC   000466            MP$RTM   001611
MO$DEF   000000            MP$CCS   004406            MP$HWA   002210            MP$MXR   001630            MP$RVE   001604
MO$DPR   000000            MP$CHN   002141            MP$IAT   001322            MP$MXW   002143            MP$RVT   000156
MO$ENT   000017            MP$CIR   000144            MP$IDE   000144            MP$NAA   006335            MP$SAC   000514
MO$EVE   000100            MP$CLK   002131            MP$IDP   006327            MP$NAC   001130            MP$SAD   000543
MO$INF   000160            MP$CLN   002126            MP$IHO   000215            MP$NAP   006326            MP$SCA   000310
MO$INS   000002            MP$CLT   002211            MP$INA   002177            MP$NCT   000240            MP$SCO   000144
MO$LIN   000001            MP$CMB   002142            MP$IND   002204            MP$NET   002114            MP$SCT   002176
MO$LOG   000001            MP$CMK   000537            MP$INI   002201            MP$NLI   000765            MP$SDU   000203
MO$MIR   000031            MP$CMX   002153            MP$ITI   000776            MP$NNA   000764            MP$SDV   000160
MO$NAM   000001            MP$CND   000310            MP$LAA   000231            MP$NOD   000500            MP$SEH   004401
MO$NIC   000023            MP$CNU   001753            MP$LAN   000234            MP$NRA   006334            MP$SER   000144
MO$NOD   000000            MP$COB   000311            MP$LAR   000202            MP$NRB   000214            MP$SET   000000
MO$OFF   000001            MP$CON   002126            MP$LAS   000012            MP$NSA   001617            MP$SGZ   001644
MO$ON    000000            MP$COS   001604            MP$LBS   000203            MP$NTI   002141            MP$SID   000176
MO$OPT   000001            MP$CPF   000230            MP$LCO   000226            MP$NUM   001642            MP$SIN   000310
MO$PER   000200            MP$CPL   000232            MP$LCT   000156            MP$NVE   001274            MP$SLI   000156
MO$PRO   000002            MP$CPT   000226            MP$LHL   000232            MP$NXN=  001476            MP$SLO   000171
MO$PRS   000001            MP$CPU   000161            MP$LLE   000227            MP$OAC   000632            MP$SMX   002202
MO$REA   000200            MP$CSZ   001755            MP$LLO   004432            MP$OAN   000620            MP$SND   000500
MO$SET   000000            MP$CUS   002127            MP$LMB   002152            MP$OCO   000776            MP$SNP   006332
MO$STA   000020            MP$CVA   000540            MP$LMX   002200            MP$OHO   000214            MP$SNU   000542
MO$SUM   000000            MP$DAL   002570            MP$LNA   000144            MP$ONA   000764            MP$SOB   000524
MO$TSK   000000            MP$DCO=  001464            MP$LNO   000233            MP$ONR   002114            MP$SPA   000157
MO$VOL   000000            MP$DDT   002177            MP$LOA   000170            MP$OQL   001453            MP$SPR   000536
MO$ZER   000000            MP$DEL   001131            MP$LOG   004514            MP$OTI   000777            MP$SPS   000513
MP$ACB   000012            MP$DES   000156            MP$LOO   000620            MP$OTY   001022            MP$STA   000000
MP$ACC   000514            MP$DEV   002114            MP$LTM   001613            MP$OUS   000777            MP$STI   002140
MP$ACT   004526            MP$DFA   001320            MP$LTY   004533            MP$OVE   001010            MP$STT   002201
MP$ADD   000766            MP$DHO=  001465            MP$LWI   000230            MP$OWN   004374            MP$STY   000175
```

```
                                                    J 13
MANVTB - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:32  Page 6-5
Symbol table
```

I 14

```
NNN         NNN        CCCCCCCCCCCC    PPPPPPPPPPPP       MMM         MMM        AAAAAAAAA       PPPPPPPPPPPP
NNN         NNN        CCCCCCCCCCCC    PPPPPPPPPPPP       MMM         MMM        AAAAAAAAA       PPPPPPPPPPPP
NNN         NNN        CCCCCCCCCCCC    PPPPPPPPPPPP       MMM         MMM        AAAAAAAAA       PPPPPPPPPPPP
NNN         NNN     CCC                PPP         PPP    MMMMMM   MMMMMM     AAA         AAA    PPP         PPP
NNN         NNN     CCC                PPP         PPP    MMMMMM   MMMMMM     AAA         AAA    PPP         PPP
NNN         NNN     CCC                PPP         PPP    MMMMMM   MMMMMM     AAA         AAA    PPP         PPP
NNNNNN      NNN     CCC                PPP         PPP    MMM   MMM   MMM     AAA         AAA    PPP         PPP
NNNNNN      NNN     CCC                PPP         PPP    MMM   MMM   MMM     AAA         AAA    PPP         PPP
NNNNNN      NNN     CCC                PPP         PPP    MMM   MMM   MMM     AAA         AAA    PPP         PPP
NNN   NNN   NNN     CCC                PPPPPPPPPPPP       MMM         MMM     AAA         AAA    PPPPPPPPPPPP
NNN   NNN   NNN     CCC                PPPPPPPPPPPP       MMM         MMM     AAA         AAA    PPPPPPPPPPPP
NNN   NNN   NNN     CCC                PPPPPPPPPPPP       MMM         MMM     AAA         AAA    PPPPPPPPPPPP
NNN      NNNNNN     CCC                PPP                MMM         MMM     AAAAAAAAAAAAAAA    PPP
NNN      NNNNNN     CCC                PPP                MMM         MMM     AAAAAAAAAAAAAAA    PPP
NNN      NNNNNN     CCC                PPP                MMM         MMM     AAAAAAAAAAAAAAA    PPP
NNN         NNN     CCC                PPP                MMM         MMM     AAA         AAA    PPP
NNN         NNN     CCC                PPP                MMM         MMM     AAA         AAA    PPP
NNN         NNN     CCC                PPP                MMM         MMM     AAA         AAA    PPP
NNN         NNN        CCCCCCCCCCCC    PPP                MMM         MMM     AAA         AAA    PPP
NNN         NNN        CCCCCCCCCCCC    PPP                MMM         MMM     AAA         AAA    PPP
NNN         NNN        CCCCCCCCCCCC    PPP                MMM         MMM     AAA         AAA    PPP
```

J 14

```
**FILE**ID**ALCHIN
```

```
                                                                        I 15
ALCHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:04        PAGE 1
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE             REFERENCES

AF.INI  = 000001          #5-61         6-156
AF.OVR  = 000002          #5-61
ALCHIN    000000 RG       #6-89
A$ALI     000012          #5-61         6-128           7-246
A$BUF     000006          #5-61        *6-92            6-155
A$DES     000022          #5-61         7-204           7-227
A$FLG     000000          #5-61        *6-156          *6-159
A$LEN     000103          #5-61         6-97
A$NUM     000010          #5-61        *7-247
A$OPT     000002          #5-61        *6-91            6-103           6-115           6-123
A$PAR     000004          #5-61        *6-132          *6-136           6-160          *7-195          *7-199
A$SCO     000020          #5-61         6-129
A$TYP     000011          #5-61         6-113           6-125
CF$VR2  = ******   GX      6-134
CHKACC    000470 R         7-232        7-235           7-238          #8-278
CHKALI  = ******   GX      6-108
CHKDES    000272 R         6-142       #7-191
CHKPAR  = ******   GX      7-197
CHKSCO  = ******   GX      6-118
CURCTX  = ******   GX      6-133
C$FLAG  = ******   GX      6-134
MAPADD  = ******   GX      7-214
MAXACC  = 000020          #5-61         5-61            7-237
MAXID     000020          #5-61         5-61            7-231
MAXPAS  = 000010          #5-61         5-61            7-234
ME$ALI    000010           6-132
ME$AL2    000003           6-136
ME.FOR    177776           6-147
ME.FUN    177777           6-102
ME.MPR    177773           6-96
ME.PVA    177760           7-250
ME.UCO    177770           6-131
MF$ALL    177775           6-113
MO$CLE    000100           6-103        6-115           6-123
MP$DES    000156           7-196
MS.MOR    000002           6-150
SAVNAM  = ******   GX      7-223
TSTALI  = ******   GX      6-130
$SAVRG  = ******   GX      6-90
```

```
                                                                        J 15
ALCHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:04        PAGE 2
MACRO CROSS REFERENCE                                      CREF    04.00
```

```
                                                   .SBTTL  FINDA - FIND A MATCHING ALIAS BLOCK
                                           ;+
                                           ; **-FINDA-FIND A MATCHING ALIAS BLOCK
                                           ;
                                           ; THIS ROUTINE FINDS A BLOCK WITH A MATCHING ALIAS NAME AND SCOPE
                                           ;
                                           ; INPUTS:
                                           ;       R0 = BLOCK AT HEAD OF LIST TO BE SCANNED
                                           ;       R4 = POINTER TO CONTEXT AREA WITH:
                                           ;            A$ALI = 6 CHAR ALIAS NAME
                                           ;            A$SCO = SCOPE OF ALIAS TO FIND
                                           ;
                                           ; OUTPUTS:
                                           ;       IF CC:
                                           ;          R1 = MAPPED ADDRESS OF BLOCK
                                           ;          $PRUNM = UNMAPPED ADDRESS OF PREVIOUS BLOCK
                                           ;          $CRUNM = UNMAPPED ADDRESS OF CURRENT BLOCK
                                           ;       IF CS:
                                           ;          NO MATCH IN LIST
                                           ;
                                           ; REGISTERS:
                                           ;       R2, R3, R4, R5 ARE PRESERVED
                                           ;-

000634                                     FINDA:
000634  010067  000000G                            MOV     R0,$CRUNM               ; INITIALIZE CURRENT BLOCK
000640  016767  000000G 000000G 10$:               MOV     $CRUNM,$PRUNM           ; INITIALIZE PREVIOUS BLOCK
000646  011067  000000G                            MOV     (R0),$CRUNM             ; SET NEW CURRENT BLOCK OF NEXT ENTRY
000652  011046                                     MOV     (R0),-(SP)              ; SET UP TO MAP TO ANB
000654                                             CALL    @CEACC                  ; MAP TO ANB
000660  000261                                     SEC                             ; ASSUME FAILURE
000662  012601                                     MOV     (SP)+,R1                ; GET MAPPED VIRTUAL ADDRESS
000664  001423                                     BEQ     40$                     ; IF EQ, THERE IS NONE (NO MATCH)
000666  026461  000012  000002                     CMP     A$ALI(R4),A.NAM(R1)     ; DOES THIS NAME MATCH ?
000674  001014                                     BNE     20$                     ; IF NE, NO
000676  026461  000014  000004                     CMP     A$ALI+2(R4),A.NAM+2(R1) ; ...
000704  001010                                     BNE     20$                     ; ...
000706  026461  000016  000006                     CMP     A$ALI+4(R4),A.NAM+4(R1) ; ...
000714  001004                                     BNE     20$                     ; ...
000716  026461  000020  000010                     CMP     A$SCO(R4),A.UCB(R1)     ; DOES THE SCOPE MATCH TOO ?
000724  001402                                     BEQ     30$                     ; IF EQ, YES - MATCH
000726  010100                             20$:    MOV     R1,R0                   ; ELSE, POINT TO NEXT BLOCK
000730  000743                                     BR      10$                     ; LOOP

000732  000241                             30$:    CLC                             ; FOUND IT !
000734                                     40$:    RETURN

        000001                                     .END
```

```
                        .SBTTL  MLOAD   - Load Control for RSX-11M Systems
                ;+
                                        *** MLOAD ***
                ;
                ; Function:     To control microcode load sequences on RSX-11M systems.
                ;               This routine performs file I/O and maintains the load seq-
                ;               uence counter(s) and status.
                ;
                ; Call Format:  CALL    MLOAD
                ;
                ; Inputs:       Input file is assumed to have been opened successfully.
                ;               R4 = address of "load request" CCB
                ;
                ; Outputs:      Input file is closed, load sequence ended
                ;               R3 = load halt status
                ;               R4 = address of "halt load" CCB
                ;
                ; Registers:    R2, R3, and R5 are destroyed
                ;
                ;-

                MLOAD:  INC     BLOCK+2                 ; Update virtual block number
                        READ$   #FDB,,,#BLOCK,,,,FCSERR ; Read the next block in file
                        BCS     10$                     ; Branch if read could not be started
                        WAIT$   #FDB                    ; Wait for read to complete
                        BCS     10$                     ; Branch if disk error

                        MOV     R4,R3                   ; Copy CCB address
                        ADD     #C.BUF1+2,R3            ; Point at first CCB buffer address
                        MOV     #FCSBUF,(R3)            ; Copy buffer's APR5-biased address
                        ADD     #20000,(R3)+            ; Force address to be APR6-biased
                        MOV     FCSIOS+2,(R3)+          ; Get buffer byte count from I/O status
                        MOV     #CF.LB,(R3)             ; One buffer/CCB on RSX-11M systems
                        CALL    DSPDRV                  ; Call driver to load the microcode
                        BCC     MLOAD                   ; Branch if load sequence not finished

                10$:    CLOSE$  #FDB                    ; Close the microcode file
                        RETURN                          ; Exit

                        .IFF                            ; If system is RSX-11S ...
```

```
                                                                  J  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31          PAGE 5

SYMBOL CROSS REFERENCE                                        CREF    04.00

SYMBOL   VALUE               REFERENCES

L.DLC       000003           #6-85
L.DLM       000006           #6-85
L.DLS       000010           #6-85
L.FLG       000000           #6-85
L.KRBA      000016           #6-85
L.LEN    =  000022           #6-85
L.MPF       000022           #6-85
L.NMST      000020           #6-85
L.NSTA      000014           #6-85
L.OWNR      000021           #6-85
L.UNT       000013           #6-85
MLD         000000 R         #9-269       18-989
MLDERR      000000 R         #7-147       11-607
MO$SAC   =  000016           #6-83
MO$SPR   =  000012           #6-83
MO$SSV   =  000014           #6-83
MO$25A   =  000006           #6-83
MO$25P   =  000002           #6-83
MO$25S   =  000004           #6-83
MO$29S   =  000010           #6-83
M$HIGH   =  000003           #6-83
M$3100   =  000000           #6-83
M$3101   =  000001           #6-83
M$3102   =  000002           #6-83
M$3103   =  000003           #6-83        6-83
M.KTEF   =  000002            8-202
NBUFS    =  000001           #7-122       18-953
NTUIC       000574 RG        #8-234
N$$UMR   =  000001           #4-2         6-87         7-112        7-120        8-200        9-312        9-321        9-367        9-392
                              10-516      10-553       12-657

OP$INI   =  000000           #6-83
OP$TER   =  000001           #6-83
PARERR      000073 R         #7-156       16-871
PARHD       000576 RG        #8-235       17-909
PCBADR      000124 R         #8-196      *16-856      *16-880
PDSPL       000612 RG        #8-241       10-547       12-680
PDVIDX      000160 R         #8-212      *9-403        10-543       11-602
PDVTA       000614 RG        #8-242       11-601
PH$HDE   =  000004           #6-83
PH$LOC   =  000002           #6-83
PH$MTS   =  000003           #6-83
PH$UMP   =  000000           #6-83
PH$WCS   =  000001           #6-83
PS.SYS   =  000040            17-915
PWRAST      000754 R          9-318      #12-648
P$$LAS   =  ******            16-858
P$$LUT   =  ******            7-137
P.MAIN      000012            17-922
P.NAM       000004            17-910      17-912
P.REL       000014            16-881
P.SIZE      000016            16-882
P.STAT      000030            17-915
```

```
                                                                  K  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31          PAGE 6

SYMBOL CROSS REFERENCE                                        CREF    04.00
```

```
    272 000240  000000G             MANTBL: .WORD   FTABLE                  ; MANAGEMENT FUNCTION TABLE
    273 000242  000466                      .WORD   CTXSIZ                  ; SIZE OF CONTEXT AREA (-2 FROM CONTEX)
    274 000244                      CONTEX::.BLKB   CTXSIZ                  ; LOW CORE COPY OF CURRENT CONTEXT AREA
    275 000732  000454                      .WORD   BUFSIZ                  ; SIZE OF BUFFER (-2 FROM BUFFER)
    276 000734                      BUFFER::.BLKB   BUFSIZ                  ; BUFFER FOR NETWORK MANAGEMENT MESSAGE
    277 001410                      IDBUFF::.BLKB   MAXID                   ; BUFFER FOR ID STRING
    278                                     .EVEN
    279
```

```
                                                              J  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-3
Symbol table

MC$NOR    000000       MD$DMC    000014       ME$CIR    000003       MF$TES    000022       MP$ADP    006325
MC$NPF    001616       MD$DMF    000046       ME$EXA    000000       MF$TRA    000301       MP$ADS    004406
MC$NPR    001630       MD$DMP    000022       ME$EXE    000200       MF$TRI    000021       MP$ALB    000036
MC$NRE    001200       MD$DMR    000050       ME$LIN    000001       MF$ZER    000025       MP$AMC    001640
MC$NRT    001166       MD$DMV    000042       ME$LOG    000002       ML$ALL    100000       MP$AMH    001641
MC$NVR    001642       MD$DN     000016       ME$MOD    000004       ML$CLS    000000       MP$ANB    004420
MC$P08    000000       MD$DP     000000       ME$NOD    000000       ML$CON    000001       MP$ASB    000024
MC$P11    000001       MD$DPV    000044       ME$NON    177777       ML$EXT    000001       MP$ASC    004432
MC$RFL    002046       MD$DQ     000006       ME$OBJ    000007       ML$FIL    000002       MP$AUS    000512
MC$SBU    002051       MD$DTE    000024       ME$OB2    000004       ML$FIR    000000       MP$BBT    002165
MC$SFL    002044       MD$DU     000002       ME$PRO    000005       ML$INT    000000       MP$BDF    002164
MC$SYC    000012       MD$DUP    000012       ME$SYS    000006       ML$KNO    140000       MP$BFQ    002121
MC$SYL    000036       MD$DV     000026       ME.BLO    177744       ML$MON    000003       MP$BLK    001616
MC$SYR    000050       MD$DZ     000030       ME.CON    177753       ML$SYS    000002       MP$BLO    001452
MC$SYS    000024       MD$FUL    000000       ME.CST    177765       ML$TOP    000001       MP$BMX    002176
MC$SYZ    000000       MD$HAL    000001       ME.DIS    177755       MN$UNL    000377       MP$BNP    006330
MC$TBR    001140       MD$HEL    000002       ME.DON    177600       MN$VOL=   000000       MP$BRT    001620
MC$TBS    001141       MD$KCP    000013       ME.FCO    177762       MOULS     000034RG     MP$BSA    002176
MC$TDR    002053       MD$KDP    000034       ME.FIO    177756       MO$ACC    000200       MP$BSD    002203
MC$TIM    000000       MD$KDZ    000036       ME.FOP    177763       MO$ADD    000002       MP$BSI    002200
MC$TYP    007777       MD$KL     000040       ME.FOR    177776       MO$ALA    000004       MP$BSP    006331
MC$UBU    002052       MD$KMX    000054       ME.FUN    177777       MO$ALI    000000       MP$BUF    000170
MC$UFD    002047       MD$KMY    000052       ME.GRO    177745       MO$CHA    000040       MP$BUP    006324
MC$UMR    001132       MD$LEN    000050       ME.HAR    177750       MO$CIR    000003       MP$BUS    001643
MC$UMS    001133       MD$PCL    000011       ME.IID    177767       MO$CLE    000100       MP$CAC    001750
MC$VAX    000003       MD$QNA    000005       ME.LCO    177766       MO$COU    000060       MP$CAS    001762
MC$WID    060000       MD$UNA    000001       ME.LPR    177757       MO$DAC    000000       MP$CAT    002210
MC$WIH    040000       MD$WIT    000002       ME.MPR    177773       MO$DEF    000000       MP$CCS    004406
MC$WIL    020000       MD.ABO    000017       ME.MVE    177771       MO$DPR    000000       MP$CHN    002141
MC$W08    020000       MD.ACC    000010       ME.OPE    177747       MO$ENT    000017       MP$CIR    000144
MC$W16    040000       MD.AOB    000016       ME.PLO    177751       MO$EVE    000100       MP$CLK    002131
MC$W32    060000       MD.BOB    000011       ME.PMI    177743       MO$INF    000160       MP$CLN    002126
MC$XBR    001750       MD.DIA    000007       ME.PNA    177752       MO$INS    000002       MP$CLT    002211
MC$XBS    001751       MD.DOB    000015       ME.PRI    177775       MO$LIN    000001       MP$CMB    002142
MC$XCJ    000322       MD.DUM    000002       ME.PTY    177772       MO$LOG    000001       MP$CMK    000537
MC$XCR    002260       MD.FAI    000014       ME.PVA    177760       MO$MIR    000031       MP$CMX    002153
MC$XCS    002261       MD.FNA    000001       ME.RES    177761       MO$NAM    000001       MP$CND    000310
MC$XDR    001762       MD.FOB    000006       ME.ROO    177754       MO$NIC    000023       MP$CNU    001753
MC$XDS    001763       MD.LOA    000001       ME.SIZ    177774       MO$NOD    000000       MP$COB    000311
MC$XFR    002272       MD.LSH    000020       ME.SYS    177746       MO$OFF    000001       MP$CON    002126
MC$XFS    002273       MD.NOB    000012       ME.UCO    177770       MO$ON     000000       MP$COS    001604
MC$XLJ    000323       MD.NON    177777       MF$ACT    177776       MO$OPT    000001       MP$CPF    000230
MC$XLR    002330       MD.PER    000000       MF$ADD    000000       MO$PER    000200       MP$CPL    000232
MC$XMA    000310       MD.RES    000004       MF$ADJ    177774       MO$PRO    000002       MP$CPT    000226
MC$XMC    002305       MD.ROB    000005       MF$ALL    177775       MO$PRS    000001       MP$CPU    000161
MC$XMS    002304       MD.RSH    000013       MF$BYE    000300       MO$REA    000200       MP$CSZ    001755
MC$XNR    002332       MD.SDU    000005       MF$CHA    000023       MO$SET    000000       MP$CUS    002127
MC$XRC    002316       MD.SLO    000003       MF$DUM    000020       MO$STA    000020       MP$CVA    000540
MC$XRR    002331       MD.SNA    000000       MF$EVT    000001       MO$SUM    000000       MP$DAL    002570
MC$XRS    002342       MD.TLO    000004       MF$FUP    000264       MO$TSK    000000       MP$DCO=   001464
MC$020    000002       MD.UNA    000002       MF$KNO    177777       MO$VOL    000000       MP$DDT    002177
MD$CI     000007       MD.UNR    000003       MF$LOA    000017       MO$ZER    000000       MP$DEL    001131
MD$CNA    000003       MD.UOB    000007       MF$LOO    177775       MP$ACB    000012       MP$DES    000156
MD$COU    000001       MD.VOL    000006       MF$REA    000024       MP$ACC    000514       MP$DEV    002114
MD$DA     000010       ME$ALI    000010       MF$SIG    177773       MP$ACT    004526       MP$DFA    001320
MD$DL     000004       ME$AL2    000003       MF$SPF    000302       MP$ADD    000766       MP$DHO=   001465
MD$DLV    000020       ME$ARE    000005       MF$SYS    000026       MP$ADJ    001440       MP$DIA    000173
```

```
                                                              K  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-4
Symbol table
```

```
     53                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     54                                  ;
     55                                  ; MACRO LIBRARY CALLS
     56                                  ;
     57                                          .MCALL  ASTX$S
```

```
                                                .SBTTL  ALLCTX - ALLOCATE A CONTEXT AREA
                                        ;+
                                        ; **-ALLCTX-ALLOCATE A CONTEXT AREA
                                        ;
                                        ; THIS ROUTINE ALLOCATES A HIGH-CORE CONTEXT AREA AND A WINDOW BLOCK
                                        ; FOR A USER. THE CONTEXT LIST IS SEARCHED FOR A FREE BLOCK. IF NONE
                                        ; IS FOUND THE CONTEXT POOL WILL BE USED. IF THE POOL IS NOT LARGE
                                        ; ENOUGH, THIS ROUTINE WILL ATTEMPT TO EXTEND THE TASK.
                                        ;
                                        ; INPUTS:
                                        ;       R3 = ADDRESS OF I/O PACKET
                                        ;       R5 = FUNCTION TABLE ENTRY WITH:
                                        ;               R$OPER(R5) = ADDRESS OF OPERATE ROUTINE
                                        ;       TSKLN2 = 2ND LUN WORD FROM USER TASK HEADER
                                        ;       CTXHD = LISTHEAD OF CONTEXT AREAS
                                        ;       POOLSZ = SIZE OF REMAINING CONTEXT POOL IN BYTES
                                        ;       POOLPT = ADDRESS OF FIRST FREE BYTE IN CONTEXT POOL
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CC, SUCCESS, AND
                                        ;          R4 = ADDRESS OF CONTEXT AREA ALLOCATED
                                        ;          CURCTX = ADDRESS OF CONTEXT ALLOCATED
                                        ;          NUMUSR IS INCREMENTED TO INDICATE ANOTHER ACTIVE USER
                                        ;          2ND LUN WORD POINTS TO WINDOW BLOCK
                                        ;          POOLSZ, AND POOLPT ARE UPDATED AS NECESSARY
                                        ;       IF CS, FAILURE
                                        ;
                                        ; REGISTERS:
                                        ;       R0, R1, AND R2 ARE MODIFIED
                                        ;-

000000                                  ALLCTX::
000000  012700  000000G                         MOV     #CTXHD,R0               ; POINT TO CONTEXT AREA LISTHEAD
000004  011004                          10$:    MOV     (R0),R4                 ; GET ADDRESS OF NEXT AREA
000006  001405                                  BEQ     20$                     ; IF EQ, NO MORE - TRY TO USE THE POOL
000010  005764  000000G                         TST     C$FLAG(R4)              ; ELSE, IS THIS CONTEXT IN USE ?
000014  001461                                  BEQ     40$                     ; IF EQ, NO - USE IT
000016  010400                                  MOV     R4,R0                   ; ELSE, SHUFFLE LIST POINTERS
000020  000771                                  BR      10$                     ; AND KEEP LOOKING

000022  026727  000000G 000000G 20$:    CMP     POOLSZ,#C$SIZE          ; IS THE POOL LARGE ENOUGH FOR A CTX ?
000030  103037                                  BHIS    30$                     ; IF HIS, YES - NO NEED TO EXTEND
000032  012701  000000G                         MOV     #C$SIZE,R1              ; SET LENGTH OF CONTEXT BLOCK
000036  016704  000000G                         MOV     POOLPT,R4               ; GET ADDRESS OF BLOCK
000042  060104                                  ADD     R1,R4                   ; CALC ENDING ADDRESS
000044  022704  160000                          CMP     #160000,R4              ; IS IT UNDER THE BLOCK?
000050  101002                                  BHI     25$                     ; IF YES, CONTINUE WITH EXTEND
000052  000261                                  SEC                             ; ELSE, SET CARRY AND REFUSE
000054  000502                                  BR      60$                     ; TO EXTEND TO APR 7
000056  166701  000000G         25$:    SUB     POOLSZ,R1               ; CALC LENGTH OF EXTEND IN BYTES
000062  062701  000077                          ADD     #63.,R1                 ; ROUND UP TO NEXT BLOCK SIZE
000066  042701  000077                          BIC     #63.,R1                 ; ...
000072  010102                                  MOV     R1,R2                   ; SAVE NUMBER OF BYTES
000074  006201                                  ASR     R1                      ; DIVIDE BY 64
000076  006201                                  ASR     R1                      ; ... TO CALC
000100  006201                                  ASR     R1                      ; ... NUMBER OF 64. BYTE BLOCKS
000102  006201                                  ASR     R1                      ; ...
```

```
                                        .TITLE  MANBYE - PERFORM LOGOUT CLEANUP FOR A USER
                                        .IDENT  /V05.00/
                                ;
                                ; COPYRIGHT (C) 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ; NETWORK MANAGEMENT - PERFORM LOGOUT CLEANUP FOR A USER
                                ;
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 4.01  17-OCT-84                 RJK
                                ;       Change Symbols for Vectoring Comm Exec
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;
```

Table of contents

```
     5-   50     MACRO CALLS, LOCAL DEFINITIONS AND STORAGE
     6-   77     FORK - SAVE THREAD CONTEXT AND RE-START SCAN
     7-  151     FRKDON - CHECK FOR COMPLETED FORKS AND RE-LOAD CONTEXT
```

```
	.TITLE	MANQIO - ISSUE NETWORK MANAGEMENT QIOS
	.IDENT	/V05.00/
	.ENABL	LC
;
; COPYRIGHT (C) 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;	MANQIO - ISSUE NETWORK MANAGEMENT QIOS FOR NCP
;
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;

```

```
                                                                                     J 10
MANQIO - ISSUE NETWORK MANAGEME MACRO V05.03b Saturday 29-Jun-85 10:30  Page 7-7
Symbol table

Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:31.03
SY:MANQIO.V2,[135,134]MANQIO/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANQIO
```

```
                                                                                     K 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
    223 000366  005701                          TST     R1                      ; ANY ACCESS CONTROL INFO TO COPY?
    224 000370  001403                          BEQ     80$                     ; BR IF NO
    225 000372  112022                  70$:    MOVB    (R0)+,(R2)+             ; COPY ACCESS CONTROL INFO
    226 000374  005301                          DEC     R1                      ; MORE TO COPY?
    227 000376  003375                          BGT     70$                     ; BR IF YES
    228 000400  160302                  80$:    SUB     R3,R2                   ; CALCULATE LENGTH TO RETURN
    229 000402  012700  000001                  MOV     #MS.SUC,R0              ; SUCCESS !
    230 000406                          90$:    RETURN
    231
    232         000001                          .END
```

```
Symbol table

MO$CIR  000003     MP$BUS  001643     MP$HAD  001757     MP$MVR  001751     MP$RSV  001754
MO$CLE  000100     MP$CAC  001750     MP$HBT  002142     MP$MWN  002154     MP$RSZ  001756
MO$COU  000060     MP$CAS  001762     MP$HDD  000162     MP$MXB  002172     MP$RTI  001616
MO$DAC  000000     MP$CAT  002210     MP$HTM  001612     MP$MXC  000466     MP$RTM  001611
MO$DEF  000000     MP$CCS  004406     MP$HWA  002210     MP$MXR  001630     MP$RVE  001604
MO$DPR  000000     MP$CHN  002141     MP$IAT  001322     MP$MXW  002143     MP$RVT  000156
MO$ENT  000017     MP$CIR  000144     MP$IDE  000144     MP$NAA  006335     MP$SAC  000514
MO$EVE  000100     MP$CLK  002131     MP$IDP  006327     MP$NAC  001130     MP$SAD  000543
MO$INF  000160     MP$CLN  002126     MP$IHD  000215     MP$NAP  006326     MP$SCA  000310
MO$INS  000002     MP$CLT  002211     MP$INA  002177     MP$NCT  000240     MP$SCD  000144
MO$LIN  000001     MP$CMB  002142     MP$IND  002204     MP$NET  002114     MP$SCT  002176
MO$LOG  000001     MP$CMK  000537     MP$INI  002201     MP$NLI  000765     MP$SDU  000203
MO$MIR  000031     MP$CMX  002153     MP$ITI  000776     MP$NNA  000764     MP$SDV  000160
MO$NAM  000001     MP$CND  000310     MP$LAA  000231     MP$NOD  000500     MP$SEH  004401
MO$NIC  000023     MP$CNU  001753     MP$LAN  000234     MP$NRA  006334     MP$SER  000144
MO$NOD  000000     MP$COB  000311     MP$LAR  000202     MP$NRB  000214     MP$SET  000000
MO$OFF  000001     MP$CON  002126     MP$LAS  000012     MP$NSA  001617     MP$SGZ  001644
MO$ON   000000     MP$COS  001604     MP$LBS  000203     MP$NTI  002141     MP$SID  000176
MO$OPT  000001     MP$CPF  000230     MP$LCD  000226     MP$NUM  001642     MP$SIN  000310
MO$PER  000200     MP$CPL  000232     MP$LCT  000156     MP$NVE  001274     MP$SLI  000156
MO$PRO  000002     MP$CPT  000226     MP$LHL  000232     MP$NXN= 001476     MP$SLO  000171
MO$PRS  000001     MP$CPU  000161     MP$LLE  000227     MP$OAC  000632     MP$SMX  002202
MO$REA  000200     MP$CSZ  001755     MP$LLO  004432     MP$OAN  000620     MP$SND  000500
MO$SET  000000     MP$CUS  002127     MP$LMB  002152     MP$OCD  000776     MP$SNP  006332
MO$STA  000020     MP$CVA  000540     MP$LMX  002200     MP$OHO  000214     MP$SNU  000542
MO$SUM  000000     MP$DAL  002570     MP$LNA  000144     MP$ONA  000764     MP$SOB  000524
MO$TSK  000000     MP$DCO= 001464     MP$LNO  000233     MP$ONR  002114     MP$SPA  000157
MO$VOL  000000     MP$DDT  002177     MP$LOA  000170     MP$OQL  001453     MP$SPR  000536
MO$ZER  000000     MP$DEL  001131     MP$LOG  004514     MP$OTI  000777     MP$SPS  000513
MP$ACB  000012     MP$DES  000156     MP$LOO  000620     MP$OTY  001022     MP$STA  000000
MP$ACC  000514     MP$DEV  002114     MP$LTM  001613     MP$OUS  000777     MP$STI  002140
MP$ACT  004526     MP$DFA  001320     MP$LTY  004533     MP$OVE  001010     MP$STT  002201
MP$ADD  000766     MP$DHO= 001465     MP$LWI  000230     MP$OWN  004374     MP$STY  000175
MP$ADJ  001440     MP$DIA  000173     MP$MAC  004420     MP$PAR  000036     MP$SUB  000001
MP$ADP  006325     MP$DLB  002571     MP$MAD  001630     MP$PAS  000513     MP$SUR  000156
MP$ADS  004406     MP$DLG  004521     MP$MAP  006323     MP$PCH  002152     MP$SUS  000512
MP$ALB  000036     MP$DLI= 001466     MP$MAR  001635     MP$PCO  000024     MP$SVR  000163
MP$AMC  001640     MP$DLT  002200     MP$MAV  006321     MP$PCT  000144     MP$SWI  002260
MP$AMH  001641     MP$DRD  001441     MP$MBN  001636     MP$PDT  002114     MP$SYP  002424
MP$ANB  004420     MP$DST  000454     MP$MBR  001637     MP$PHA  000170     MP$TFL  000156
MP$ASB  000024     MP$DTE  002140     MP$MBU  001642     MP$PHY  000012     MP$TH1  002206
MP$ASC  004432     MP$DTY= 001452     MP$MCB  000156     MP$PLI  000025     MP$TH2  002205
MP$AUS  000512     MP$DUA  000207     MP$MCO  001632     MP$PLN  002140     MP$TH3  002202
MP$BBT  002165     MP$DUC  000210     MP$MDE  004421     MP$PLO  000012     MP$TLN  000202
MP$BDF  002164     MP$DUM  000202     MP$MHO  001633     MP$PMC  004374     MP$TLO  000172
MP$BFQ  002121     MP$DUP  002127     MP$MLB  000202     MP$PNT  002126     MP$TPA  004375
MP$BLK  001616     MP$DVC  002114     MP$MLK  001306     MP$PRI  004411     MP$TRI  002164
MP$BLO  001452     MP$DWE  001321     MP$MLN  001631     MP$PRO  002130     MP$TST  000144
MP$BMX  002176     MP$ELT  000157     MP$MLP  006333     MP$PSS  001763     MP$TYP  002130
MP$BNP  006330     MP$ETY  001605     MP$MRB  002171     MP$PST  001762     MP$UCS  004407
MP$BRT  001620     MP$EVE  000311     MP$MRO  001605     MP$RET  001631     MP$USR  001750
MP$BSA  002176     MP$FNC  001752     MP$MRP  006322     MP$RFA  001323     MP$VEC  004410
MP$BSD  002203     MP$GDT  002222     MP$MRT  002153     MP$RMX  002201     MP$VER  004406
MP$BSI  002200     MP$GNM  002223     MP$MRV  006320     MP$RPA  004374     MP$WDF  002165
MP$BSP  006331     MP$GRO  002115     MP$MSB  000170     MP$RPR  001606     MP$WMX  002177
MP$BUF  000170     MP$GRP  000541     MP$MVE  000145     MP$RRT  002213     MP$XMT  002166
MP$BUP  006324     MP$GTY  002224     MP$MVI  001634     MP$RST  002212     MP$XPF  000242
```

```
Symbol table
```

```
                                                                                J 13
MANVTB - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:32  Page 6-5
Symbol table

MP$SUB  000001          MS$GLO  000000          MT$HEX  000040          MX$NMS  000454          OBCHIN= ****** GX
MP$SUR  000156          MS$HOL  000002          MT$LPB  000005          MX$NOD  000006          OBCHOP= ****** GX
MP$SUS  000512          MS$INA  000002          MT$MAX  000037          MX$OBJ  000006          OBREIN= ****** GX
MP$SVR  000163          MS$LOA  000003          MT$MUL  000100          MX$OWN  000040          OBREOP= ****** GX
MP$SWI  002260          MS$LOO  000002          MT$NLE  000017          MX$PAR  000036          PRCHIN= ****** GX
MP$SYP  002424          MS$OFF  000001          MT$NON  000001          MX$PAS  000010          PRCHOP= ****** GX
MP$TFL  000156          MS$ON   000000          MT$NR4  000005          MX$RAC  000047          PRREIN= ****** GX
MP$TH1  002206          MS$PRO  000006          MT$NTY  000060          MX$RID  000047          PRREOP= ****** GX
MP$TH2  002205          MS$REA  000004          MT$OCT  000060          MX$RPS  000047          P$$P45= 000000
MP$TH3  002202          MS$REF  000001          MT$PHA  000002          MX$SID  000040          P$$WRD= 000000
MP$TLN  000202          MS$RES  000003          MT$POI  000000          MX$SNK  000377          Q$$OPT= 000010
MP$TLO  000172          MS$RST  000001          MT$QP2  000010          MX$TYP  000077          RF$END= ****** GX
MP$TPA  004375          MS$RSX  000002          MT$ROU  000000          MX$UID  000020          RF$NCX= ****** GX
MP$TRI  002164          MS$RT   000005          MT$RO4  000004          M$$CRB= 000124          RF$PRV= ****** GX
MP$TST  000144          MS$SER  000002          MT$SEC  000000          M$$CRX= 000000          R$$DER= 000000
MP$TYP  002130          MS$SHU  000002          MT$SGD  000020          M$$FCS= 000000          R$$K11= 000001
MP$UCS  004407          MS$SRV  000007          MT$SYS  000002          M$$MGE= 000000          R$$SND= 000000
MP$USR  001750          MS$STA  000000          MT$TER  000001          M$$NET= 000000          R$$11M= 000000
MP$VEC  004410          MS$SYN  000012          MT$TRI  000002          M$$OVR= 000000          SF$ANM  000001
MP$VER  004406          MS$1ER  000001          MT$TYP  007777          NM.CLN  003400          SF$DNM  000002
MP$WDF  002165          MS$TOP  000003          MT$USD  000000          NM.INI  002400          SYCHIN= ****** GX
MP$WMX  002177          MS$TRI  000005          MT$X25  000003          NM.OPR  003000          SYCHOP= ****** GX
MP$XMT  002166          MS$UNR  000005          MU$INC  000001          NM.VR2  002401          SYREIN= ****** GX
MP$XPF  000242          MS$VMS  000004          MU$OUT  000002          NM.VR3  002402          SYREOP= ****** GX
MP$XPL  000244          MS.DON  177600          MU$PER  000000          NOCHIN= ****** GX       SYZEIN= ****** GX
MP$XPT  000240          MS.MOR  000002          MV$III  000000          NOCHOP= ****** GX       SYZEOP= ****** GX
MP$XXX  177777          MS.PAR  000003          MV$IV   000001          NOREIN= ****** GX       S$$WRG= 000000
MP$5ST  004406          MS.SUC  000001          MX$ACT  000020          NOREOP= ****** GX       S$$YSZ= 007600
MS$ACT  000001          MT$AR4  000003          MX$CIR  000020          NOZEIN= ****** GX       TRAACC= ****** GX
MS$ADU  000010          MT$ASC  000100          MX$CLN  000020          NOZEOP= ****** GX       TRAADD= ****** GX
MS$ALO  000007          MT$BIL  000001          MX$CNM  000006          N$$ACC= 000001          TRAALI= ****** GX
MS$ASE  000006          MT$BYS  000011          MX$CON  000006          N$$BUF= 000001          TRANAM= ****** GX
MS$ATR  000011          MT$CI   000007          MX$DAC  000020          N$$LDV= 000001          T$$KMG= 000000
MS$AUT  000000          MT$CLE  000077          MX$DTE  000020          N$$MCP= 000001          T$$MIN= 000000
MS$CLE  000003          MT$COD  000200          MX$FIL  000034          N$$MLL= 000001          V$$CTR= 001000
MS$DED  000004          MT$CON  000001          MX$LGC  000777          N$$MOV= 000010          X$$DBT= 000000
MS$DIE  000003          MT$DMC  000004          MX$LIN  000020          N$$NCT= 000001          $$$SYS= 004374
MS$DUM  000004          MT$ETH  000006          MX$LON  000006          N$$PEM= 000001          $$$SYX= 000000
MS$FAI  000013

. ABS.  000000      000   (RW,I,GBL,ABS,OVR)
        000000      001   (RW,I,LCL,REL,CON)
DATA$   000470      002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 191
Work  file writes: 125
Size of work file: 25093 Words  ( 99 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:24.01
SY:MANVTB.V2,[135,134]MANVTB/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANVTB
```

```
                                                                  K 13
MANVTB      CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
**FILE**ID**ALCHIN
```

```
  AAAAAA    LL            CCCCCCCC  HH      HH    IIIIII    NN      NN
  AAAAAA    LL            CCCCCCCC  HH      HH    IIIIII    NN      NN
AA      AA  LL          CC          HH      HH      II      NN      NN
AA      AA  LL          CC          HH      HH      II      NN      NN
AA      AA  LL          CC          HH      HH      II      NNNN    NN
AA      AA  LL          CC          HH      HH      II      NNNN    NN
AA      AA  LL          CC          HHHHHHHHHH      II      NN  NN  NN
AA      AA  LL          CC          HHHHHHHHHH      II      NN  NN  NN
AAAAAAAAAA  LL          CC          HH      HH      II      NN    NNNN
AAAAAAAAAA  LL          CC          HH      HH      II      NN    NNNN
AA      AA  LL          CC          HH      HH      II      NN      NN      ....
AA      AA  LL          CC          HH      HH      II      NN      NN      ....
AA      AA  LLLLLLLLLL    CCCCCCCC  HH      HH    IIIIII    NN      NN      ....
AA      AA  LLLLLLLLLL    CCCCCCCC  HH      HH    IIIIII    NN      NN      ....
```

```
LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

```
ALCHIN - ALIAS CHANGE PARAMETER MACRO V05.03b Saturday 29-Jun-85 12:04
Table of contents
```

J 15

```
ALCHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:04     PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

ALCCX$           #5-57        5-61
CALL              6-108       6-118       6-130       6-142       7-197       7-214       7-223       7-232       7-235       7-238
MANDF$           #5-57        5-59
RETURN            6-165       7-252       8-297
```

K 15

```
**FILE**ID**ALCHOP
```

```
                                                                 J 16
ALCHOP - ALIAS CHANGE PARAMETER MACRO V05.03b Tuesday 10-Sep-85 11:45  Page 10-1
Symbol table

AF.INI= 000001      D$MLL   000040      H.X25   000064      MC$LUP  006072      MC$XFS  002273
AF.OVR= 000002      D$MNOD  000041      I$$RAR= 000000      MC$MAP  010000      MC$XLJ  000323
ALCHOP  000000RG    D$NA    000062      I$$RDN= 000000      MC$MBL  001764      MC$XLR  002330
ALOCB = ****** GX   D$NBEA  000056      KISAR6= ****** GX   MC$MBX  004420      MC$XMA  000310
A$ALI   000012      D$NBRA  000054      K$$CNT= 177546      MC$MBY  001752      MC$XMC  002305
A$BUF   000006      D$NEND= 000054      K$$CSR= 177546      MC$MPX  004406      MC$XMS  002304
A$DES   000022      D$NLN   000030      K$$LDC= 000000      MC$NAP  001604      MC$XNR  002332
A$FLG   000000      D$NN    000060      K$$TPS= 000074      MC$NBR  001130      MC$XRC  002316
A$LEN   000103      D$OUTT  000043      LC$EXT  000000      MC$NBS  001131      MC$XRR  002331
A$NUM   000010      D$RETF  000050      LC$INT  000001      MC$NCR  001154      MC$XRS  002342
A$OPT   000002      D$RNN   000002      LD$LP = 000000      MC$NCS  001155      MC$020  000002
A$PAR   000004      D$RTMR  000076      L$$ASG= 000000      MC$NML  001274      MD$CI   000007
A$SCO   000020      D$SEG   000036      L$$DRV= 000000      MC$NMR  001142      MD$CNA  000003
A$TYP   000011      D$SER   000032      L$$P11= 000001      MC$NMS  001143      MD$COU  000001
A$$CHK= 000000      D$SQRL  000052      L$$11R= 000000      MC$NND  001606      MD$DA   000010
A$$CPS= 000000      D$$BUG= 177514      MAXACC= 000020      MC$NNU  001605      MD$DL   000004
A$$PRI= 000000      D$$ISK= 000000      MAXID = 000020      MC$NOP  001607      MD$DLV  000020
A$$TRP= 000000      D$$L11= 000001      MAXPAS= 000010      MC$NOR  000000      MD$DMC  000014
A.ACC   000020      D$$YNC= 000000      MB$DIS  000001      MC$NPF  001616      MD$DMF  000046
A.LNK   000000      D$$YNM= 000000      MB$ENA  000000      MC$NPR  001630      MD$DMP  000022
A.NAM   000002      E$$XPR= 000000      MB$FUL  000002      MC$NRE  001200      MD$DMR  000050
A.PUD   000010      FINDA   000634R     MB$MIX  000002      MC$NRT  001166      MD$DMV  000042
A.REM   000012      F$$LVL= 000001      MB$ONE  000001      MC$NVR  001642      MD$DN   000016
A.UCB   000010      G$$TPP= 000000      MB$RXO  000001      MC$P08  000000      MD$DP   000000
BIAS  = 000002      G$$TSS= 000000      MB$TXO  000000      MC$P11  000001      MD$DPV  000044
CEACC = ****** GX   G$$TTK= 000000      MB$ZER  000000      MC$RFL  002046      MD$DQ   000006
CF$VR2= ****** GX   G$$WRD= 000000      MC$BAB  006073      MC$SBU  002051      MD$DTE  000024
CLRALI  000336R     H.CSP   000000      MC$BID  001765      MC$SFL  002044      MD$DU   000002
CURCTX= ****** GX   H.CUIC  000010      MC$BMC  001767      MC$SYC  000012      MD$DUP  000012
C$FLAG= ****** GX   H.DSW   000046      MC$BSC  001766      MC$SYL  000036      MD$DV   000026
C$$CKP= 000000      H.DUIC  000012      MC$CAC  001442      MC$SYR  000050      MD$DZ   000030
C$$ORE= 000400      H.EFLM  000004      MC$CAP  001440      MC$SYS  000024      MD$FUL  000000
C$$RSH= 177564      H.EFSV  000040      MC$CCL  001445      MC$SYZ  000000      MD$HAL  000001
DEAANB  000534R     H.FCS   000050      MC$CDC  002045      MC$TBR  001140      MD$HEL  000002
DEACB = ****** GX   H.FORT  000052      MC$CDP  001441      MC$TBS  001141      MD$KCP  000013
DECPT = ****** GX   H.FPSA  000042      MC$CIF  001465      MC$TDR  002053      MD$KDP  000034
D$AMXC  000072      H.FPSL= 000062      MC$CLD  001464      MC$TIM  000000      MD$KDZ  000036
D$AMXH  000074      H.FPVA  000034      MC$COU  100000      MC$TYP  007777      MD$KL   000040
D$ANN   000000      H.GARD  000072      MC$CTL  001454      MC$UBU  002052      MD$KMX  000054
D$BRPR  000102      H.HDLN  000002      MC$CTR  001452      MC$UFD  002047      MD$KMY  000052
D$BRTM  000100      H.IPC   000016      MC$CTS  001453      MC$UMR  001132      MD$LEN  000050
D$DELF  000045      H.IPS   000014      MC$DOV  002050      MC$UMS  001133      MD$PCL  000011
D$DELW  000046      H.ISP   000020      MC$LBR  001750      MC$VAX  000003      MD$QNA  000005
D$END = 000104      H.LUN   000076      MC$LBS  001751      MC$WID  060000      MD$UNA  000001
D$FNB   000034      H.NLUN  000074      MC$LDI  001774      MC$WIH  040000      MD$WIT  000002
D$HIOR  000024      H.NML   000061      MC$LDO  001775      MC$WIL  020000      MD.ABO  000017
D$HOST  000022      H.ODVA  000022      MC$LDR  001762      MC$W08  020000      MD.ACC  000010
D$INAC  000044      H.ODVL  000024      MC$LDS  001763      MC$W16  040000      MD.AOB  000016
D$INCT  000042      H.OVLY  000054      MC$LLB  002021      MC$W32  060000      MD.BOB  000011
D$IPL   000051      H.PFVA  000032      MC$LLP  002115      MC$XBR  001750      MD.DIA  000007
D$LID   000020      H.RCVA  000036      MC$LLR  002007      MC$XBS  001751      MD.DOB  000015
D$LNAM  000006      H.RRVA  000062      MC$LOO  000001      MC$XCJ  000322      MD.DUM  000002
D$LNUM  000014      H.SPRI  000060      MC$LRB  002020      MC$XCR  002260      MD.FAI  000014
D$LST   000047      H.TKVA  000026      MC$LRP  002114      MC$XCS  002261      MD.FNA  000001
D$MAXC  000064      H.TKVL  000030      MC$LRR  002006      MC$XDR  001762      MD.FOB  000006
D$MAXH  000066      H.VEXT  000056      MC$LSI  002032      MC$XDS  001763      MD.LOA  000001
D$MAXV  000070      H.WND   000044      MC$LST  002033      MC$XFR  002272      MD.LSH  000020
```

```
                                                                 K 16
ALCHOP - ALIAS CHANGE PARAMETER MACRO V05.03b Tuesday 10-Sep-85 11:45  Page 10-2
Symbol table
```

```
                                        .SBTTL  LOCPAR  - Locate Microcode Partition
                                ;+
                                                        *** LOCPAR ***
                                ;
                                ; Function:     To locate a (sub)partition that contains the microcode
                                ;               specified in the current load request.  If the partition
                                ;               is found,  it's PCB address is saved for later use.
                                ;
                                ; Call Format:  CALL    LOCPAR
                                ;
                                ; Inputs:       R4 = address of microcode partition ASCII name
                                ;               R5 = length of ASCII string
                                ;
                                ; Outputs:      C-bit clear = partition found
                                ;               C-bit set   = partition not found
                                ;
                                ;               If the partition has been located, then:
                                ;
                                ;               BLOCK   = mapping bias of partition
                                ;               BLOCK+2 = length of partition in 32-word blocks
                                ;
                                ; Registers:    R0 through R3 destroyed
                                ;
                                ;-
001074  010403                  LOCPAR: MOV     R4,R3                   ; Copy address of string
001076  060503                          ADD     R5,R3                   ; Point past end of string
001100  111346                          MOVB    (R3),-(SP)              ; Save whatever is there
001102  112713  000011                  MOVB    #11,(R3)                ; Put a known delimiter in its place
001106  010346                          MOV     R3,-(SP)                ; Save the delimiter address
001110  005046                          CLR     -(SP)                   ; Allocate two words on the stack
001112  005046                          CLR     -(SP)                   ;   to store RAD50 partition name

001114  012701  000001                  MOV     #1,R1                   ; Allow periods to be in partition name
001120  010400                          MOV     R4,R0                   ; Copy address of ASCII string
001122                                  CALL    $CAT5                   ; Convert three characters to RAD50
001126  010116                          MOV     R1,(SP)                 ; Move RAD50 equivalent to save area
001130  103406                          BCS     10$                     ; Branch if less than three converted
001132  012701  000001                  MOV     #1,R1                   ; Allow periods to be in partition name
001136                                  CALL    $CAT5                   ; Convert three more characters
001142  010166  000002                  MOV     R1,2(SP)                ; Move RAD50 equivalent to save area

001146  010603                  10$:    MOV     SP,R3                   ; Copy address of RAD50 partition name
001150  005067  000124'                 CLR     PCBADR                  ; Clear local PCB address

                                        .IF     DF      P$$LAS          ; If system has PLAS support ...

                                        CALL    $SRNAM                  ; Use executive routine to find PCB

                                        .IFF                            ; If no PLAS support ...

001154                                  CALL    SRNAM                   ; Use internal routine to find PCB

                                        .IFTF                           ; Regardless of PLAS support ...

001160  103016                          BCC     20$                     ; Branch if partition found

```

```
                                                                     K  2
MLD11S      CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 6
SYMBOL CROSS REFERENCE                                    CREF    04.00
SYMBOL   VALUE              REFERENCES

P.SUB      000010             17-917       17-919
QRMVF      000600 RG        #8-236         9-361        12-670
RDBSZ      000616 RG        #8-243         10-522       10-524       10-558       10-559
RE$ADC   = 000004           #6-83
RE$ADF   = 000017           #6-83
RE$ADR   = 000007           #6-83
RE$BLK   = 000010           #6-83
RE$CAF   = 000014           #6-83
RE$DAT   = 000001           #6-83
RE$DRP   = 000016           #6-83
RE$LDT   = 000013           #6-83
RE$LSN   = 000012           #6-83
RE$NML   = 000001           #6-83
RE$OPE   = 000004           #6-83
RE$OPR   = 000000           #6-83
RE$RCV   = 000001           #6-83
RE$SED   = 000011           #6-83
RE$SKW   = 000006           #6-83
RE$STA   = 000002           #6-83
RE$SUM   = 000003           #6-83
RE$SYN   = 000000           #6-83
RE$TME   = 000021           #6-83
RE$TMO   = 000000           #6-83
RE$TMR   = 000020           #6-83
RE$UPT   = 000002           #6-83
RE$URE   = 000003           #6-83
RE$VER   = 000005           #6-83
RE$VRQ   = 000015           #6-83
RT$INI   = 000002           #6-83
RT$OFF   = 000001           #6-83
RT$ON    = 000000           #6-83
R$$MPL   = ******             6-54         8-231        9-407
R$$11D   = ******             6-84
R$$11M   = 000000             6-84
R$$11S   = 000001           #4-3           6-62         6-84
SC$OFF   = 000001           #6-83
SC$ON    = 000000           #6-83
SC$RST   = 000003           #6-83
SC$SHU   = 000002           #6-83
SETTMR     000126 R         #8-202         9-340
SF.ACT   = 000200           #6-85
SF.ENA   = 000100           #6-85
SF.LPB   = 000004           #6-85
SF.MFL   = 000040           #6-85
SF.PAC   = 000020           #6-85
SF.REA   = 000010           #6-85
SF.SER   = 000001           #6-85
SF.SVC   = 000002           #6-85
SF.UNL   = 000040           #6-85
SLOAD      001330 R           9-455       #18-953       18-983
SLTMA      000620 RG        #8-244          9-400
SRNAM      001246 R           16-864      #17-909
```

```
                                                                     L  2
MLD11S      CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 7
SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
                                        .SBTTL  MANACP - NETWORK MANAGEMENT ACP
                                        .SBTTL  GETPKT - GET NEXT I/O PACKET (ACP RE-START AFTER FORK)
                                ;+
                                ; **-MANACP-NETWORK MANAGEMENT ACP FUNCTION DISPATCHING
                                ;
                                ; THIS ROUTINE IS THE MAIN LINE OF THE NETWORK MANAGEMENT ACP. IT
                                ; PROCESSES I/O PACKETS UNTIL THE RECEIVE QUEUE IS EMPTY. WHEN THERE
                                ; ARE NO PACKETS TO PROCESS, ANY THREADS WHICH HAVE FORKED AND ARE
                                ; READY TO BE RESUMED WILL BE PROCESSED. IF THE NUMBER OF CURRENT
                                ; USERS IS 0, THE TASK WILL EXIT, ELSE IT WILL STOP.
                                ;
                                ; EACH I/O PACKET IS DISPATCHED TO A PROCESSING ROUTINE ACCORDING
                                ; TO ITS FUNCTION (I.FCN+1).
                                ;
                                ; INPUTS:
                                ;       I/O PACKETS QUEUED FROM T.RCVL
                                ;
                                ; OUTPUTS:
                                ;       $IOFIN CALLED TO COMPLETE I/O
                                ;-

001472                          MANACP::
                                ;
                                ; INITIALIZE THE ACP
                                ;
001472  010667  176530                  MOV     SP,ISTACK               ; SAVE INTIAL STACK POINTER
001476                                  CALL    INIACP                  ; INITIALIZE ACP
                                ;
                                ; GET AN I/O PACKET, IF NONE, CHECK FOR COMPLETED FORKS, IF NONE,
                                ; CHECK FOR CURRENT USERS. IF USERS <> 0, STOP TASK, ELSE EXIT

001502                          GETPKT::SWSTK$  50$                     ;; ENTER SYSTEM STATE
001506  005067  176476          10$:    CLR     CURIOP                  ;; ASSUME NO I/O PACKET
001512  005067  176474                  CLR     CURCTX                  ;; NO CURRENT CONTEXT
001516  005067  176474                  CLR     DELFLG                  ;; DON'T DELETE CURRENT CONTEXT (YET)
001522  017700  176316                  MOV     @TKTCB,R0               ;; POINT TO OUR TCB             ; RJK02

                                        .IF DF  MN$VOL                  ;; ONLY DONE IN THE VOLATILE ACP
                                ;
                                ; LOOK FOR PROCESS LEVEL MANAGEMENT REQUESTS WHICH HAVE BEEN COMPLETED.
                                ; THIS IS DONE BEFORE I/O PACKET PROCESSING, SO THAT SYSTEM RESOURCES
                                ; ARE RELEASED AS SOON AS POSSIBLE.
                                ;
001526  032777  000000G 176272          BIT     #FE.CEX,@FMASK          ;; COMM EXEC LOADED             ; RJK02
001534  001433                          BEQ     13$                     ;; IF EQ, NO - SKIP CHECKING LIST
001536  016703  176356                  MOV     NMCLH,R3                ;; GET POINTER TO LISTHEAD      ; RJK02
001542  010302                  11$:    MOV     R3,R2                   ;; SAVE UNLINK ADDRESS
001544  011303                          MOV     (R3),R3                 ;; POINT TO NEXT CCB
001546  001426                          BEQ     13$                     ;; IF EQ, NO COMPLETED PROCESS LEVEL REQUESTS
001550  026063  000000G 000034          CMP     T.NAM(R0),C.ADD(R3)     ;; COMPARE FIRST PART OF TASKNAME
001556  001371                          BNE     11$                     ;; IF NE, IT'S NOT OUR CCB
001560  026063  000002G 000036          CMP     T.NAM+2(R0),C.ADD+2(R3) ;; COMPARE SECOND PART OF TASKNAME
001566  001365                          BNE     11$                     ;; IF NE, IT'S SOMEONE ELSES
001570  011312                          MOV     (R3),(R2)               ;; ELSE, UNLINK BUFFER FROM LIST
001572  001002                          BNE     12$                     ;; IF NE, NOT LAST IN LIST
001574  010277  176322                  MOV     R2,@NMCL2               ;; ELSE, MODIFY LIST TAIL POINTER; RJK02
001600  005267  176420          12$:    INC     FDNCNT                  ;; INDICATE A FORK HAS COMPLETED
```

```
                                                         K  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-4
Symbol table

MP$DLB  002571          MP$MAD  001630          MP$PAS  000513          MP$SUR  000156          MS$TER  000001
MP$DLG  004521          MP$MAP  006323          MP$PCH  002152          MP$SUS  000512          MS$TOP  000003
MP$DLI= 001466          MP$MAR  001635          MP$PCO  000024          MP$SVR  000163          MS$TRI  000005
MP$DLT  002200          MP$MAV  006321          MP$PCT  000144          MP$SWI  002260          MS$UNR  000005
MP$DRO  001441          MP$MBN  001636          MP$PDT  002114          MP$SYP  002424          MS$VMS  000004
MP$DST  000454          MP$MBR  001637          MP$PHA  000170          MP$TFL  000156          MS.DON  177600
MP$DTE  002140          MP$MBU  001642          MP$PHY  000012          MP$TH1  002206          MS.MOR  000002
MP$DTY= 001452          MP$MCB  000156          MP$PLI  000025          MP$TH2  002205          MS.PAR  000003
MP$DUA  000207          MP$MCO  001632          MP$PLN  002140          MP$TH3  002202          MS.SUC  000001
MP$DUC  000210          MP$MDE  004421          MP$PLO  000012          MP$TLN  000202          MT$AR4  000003
MP$DUM  000202          MP$MHO  001633          MP$PMC  004374          MP$TLO  000172          MT$ASC  000100
MP$DUP  002127          MP$MLB  000202          MP$PNT  002126          MP$TPA  004375          MT$BIL  000001
MP$DVC  002114          MP$MLK  001306          MP$PRI  004411          MP$TRI  002164          MT$BYS  000011
MP$DWE  001321          MP$MLN  001631          MP$PRO  002130          MP$TST  000144          MT$CI   000007
MP$ELT  000157          MP$MLP  006333          MP$PSS  001763          MP$TYP  002130          MT$CLE  000077
MP$ETY  001605          MP$MRB  002171          MP$PST  001762          MP$UCS  004407          MT$COD  000200
MP$EVE  000311          MP$MRO  001605          MP$RET  001631          MP$USR  001750          MT$CON  000001
MP$FNC  001752          MP$MRP  006322          MP$RFA  001323          MP$VEC  004410          MT$DMC  000004
MP$GDT  002222          MP$MRT  002153          MP$RMX  002201          MP$VER  004406          MT$ETH  000006
MP$GNM  002223          MP$MRV  006320          MP$RPA  004374          MP$WDF  002165          MT$HEX  000040
MP$GRO  002115          MP$MSB  000170          MP$RPR  001606          MP$WMX  002177          MT$LPB  000005
MP$GRP  000541          MP$MVE  000145          MP$RRT  002213          MP$XMT  002166          MT$MAX  000037
MP$GTY  002224          MP$MVI  001634          MP$RST  002212          MP$XPF  000242          MT$MUL  000100
MP$HAD  001757          MP$MVR  001751          MP$RSV  001754          MP$XPL  000244          MT$NLE  000017
MP$HBT  002142          MP$MWN  002154          MP$RSZ  001756          MP$XPT  000240          MT$NON  000001
MP$HDD  000162          MP$MXB  002172          MP$RTI  001616          MP$XXX  177777          MT$NR4  000005
MP$HTM  001612          MP$MXC  000466          MP$RTM  001611          MP$5ST  004406          MT$NTY  000060
MP$HWA  002210          MP$MXR  001630          MP$RVE  001604          MS$ACT  000001          MT$OCT  000060
MP$IAT  001322          MP$MXW  002143          MP$RVT  000156          MS$ADU  000010          MT$PHA  000002
MP$IDE  000144          MP$NAA  006335          MP$SAC  000514          MS$ALO  000007          MT$POI  000000
MP$IDP  006327          MP$NAC  001130          MP$SAD  000543          MS$ASE  000006          MT$QP2  000010
MP$IHO  000215          MP$NAP  006326          MP$SCA  000310          MS$ATR  000011          MT$ROU  000000
MP$INA  002177          MP$NCT  000240          MP$SCO  000144          MS$AUT  000000          MT$R04  000004
MP$IND  002204          MP$NET  002114          MP$SCT  002176          MS$CLE  000003          MT$SEC  000000
MP$INI  002201          MP$NLI  000765          MP$SDU  000203          MS$DED  000004          MT$SGD  000020
MP$ITI  000776          MP$NNA  000764          MP$SDV  000160          MS$DIE  000003          MT$SYS  000002
MP$LAA  000231          MP$NOD  000500          MP$SEH  004401          MS$DUM  000004          MT$TER  000001
MP$LAN  000234          MP$NRA  006334          MP$SER  000144          MS$FAI  000013          MT$TRI  000002
MP$LAR  000202          MP$NRB  000214          MP$SET  000000          MS$GLO  000000          MT$TYP  007777
MP$LAS  000012          MP$NSA  001617          MP$SGZ  001644          MS$HOL  000002          MT$USD  000000
MP$LBS  000203          MP$NTI  002141          MP$SID  000176          MS$INA  000002          MT$X25  000003
MP$LCO  000226          MP$NUM  001642          MP$SIN  000310          MS$LOA  000003          MU$INC  000001
MP$LCT  000156          MP$NVE  001274          MP$SLI  000156          MS$LOO  000002          MU$OUT  000002
MP$LHL  000232          MP$NXN= 001476          MP$SLO  000171          MS$OFF  000001          MU$PER  000000
MP$LLE  000227          MP$OAC  000632          MP$SMX  002202          MS$ON   000000          MV$III  000000
MP$LLO  004432          MP$OAN  000620          MP$SND  000500          MS$PRO  000006          MV$IV   000001
MP$LMB  002152          MP$OCO  000776          MP$SNP  006332          MS$REA  000004          MX$ACT  000020
MP$LMX  002200          MP$OHO  000214          MP$SNU  000542          MS$REF  000001          MX$CIR  000020
MP$LNA  000144          MP$ONA  000764          MP$SOB  000524          MS$RES  000003          MX$CLN  000020
MP$LNO  000233          MP$ONR  002114          MP$SPA  000157          MS$RST  000001          MX$CNM  000006
MP$LOA  000170          MP$OQL  001453          MP$SPR  000536          MS$RSX  000002          MX$CON  000006
MP$LOG  004514          MP$OTI  000777          MP$SPS  000513          MS$RT   000005          MX$DAC  000020
MP$LOC  000620          MP$OTY  001022          MP$STA  000000          MS$SER  000002          MX$DTE  000020
MP$LTM  001613          MP$OUS  000777          MP$STI  002140          MS$SHU  000002          MX$FIL  000034
MP$LTY  004533          MP$OVE  001010          MP$STT  002201          MS$SRV  000007          MX$LGC  000777
MP$LWI  000230          MP$OWN  004374          MP$STY  000175          MS$STA  000000          MX$LIN  000020
MP$MAC  004420          MP$PAR  000036          MP$SUB  000001          MS$SYN  000012          MX$LON  000006
```

```
                                                         L  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-5
Symbol table
```

```
                                                .SBTTL  NTLAST - NTL COMPLETION AST HANDLING
                                        ;+
                                        ; **-NTLAST-NTL COMPLETION AST HANDLING
                                        ;
                                        ; THIS ROUTINE IS CALLED WHEN NTL/NTINIT HAS FINISHED PROCESSING
                                        ; A REQUEST. IT STORES THE REQUEST COMPLETION STATUS, AND MARKS
                                        ; THE THREAD AS BEING READY TO BE RESUMED.
                                        ;
                                        ; INPUTS:
                                        ;         (SP) = ADDRESS OF THREAD CONTEXT AREA
                                        ;        2(SP) = COMPLETION STATUS OF REQUEST
                                        ;
                                        ; OUTPUTS:
                                        ;       STACK CLEANED OFF, AND
                                        ;       C$STAT IN CONTEXT AREA = COMPLETION STATUS
                                        ;       CF$FDN IS SET IN C$FLAG OF CONTEXT AREA TO INDICATE FORK IS DONE
                                        ;       FDNCNT IS INCREMENTED TO INDICATE A THREAD IS READY TO BE RESUMED.
                                        ;
                                        ; REGISTERS:
                                        ;       NO REGISTERS ARE MODIFIED
                                        ;-

000000                                  NTLAST::
000000  010046                                  MOV     R0,-(SP)                ; SAVE R0
000002  016600  000002                          MOV     2(SP),R0                ; GET CONTEXT AREA ADDRESS
000006  052760  000000G 000000G                 BIS     #CF$FDN,C$FLAG(R0)      ; MARK THREAD AS RESUMEABLE
000014  016660  000004  000000G                 MOV     4(SP),C$STAT(R0)        ; STORE COMPLETION STATUS
000022  012600                          ASTX:   MOV     (SP)+,R0                ; RESTORE R0
000024  022626                          ASTX1:  CMP     (SP)+,(SP)+             ; CLEAN OFF STACK
000026                                          SWSTK$  10$                     ;; ENTER SYSTEM STATE
000032  017700  000000G                         MOV     @TKTCB,R0               ;; GET OUR TCB ADDRESS          ; RJK01
000036                                          CALLR   @EXRQN                  ;; UNSTOP OURSELF AND RETURN    ; RJK01
000042  005267  000000G                 10$:    INC     FDNCNT                  ; INDICATE A FORK IS COMPLETE
000046                                  REXAST::ASTX$S                          ; EXIT THE AST
```

```
000104  006201                          ASR     R1                      ; ...
000106  006201                          ASR     R1                      ; ...
000110                                  EXTK$S  R1                      ; TRY TO EXTEND THE TASK
000122  103457                          BCS     60$                     ; IF CS, RETURN WITH ERROR
000124  060267  000000G                 ADD     R2,POOLSZ               ; ADD EXTEND SIZE TO POOL

000130  162767  000000G 000000G 30$:    SUB     #C$SIZE,POOLSZ          ; REDUCE POOL SIZE BY A CONTEXT BLOCK
000136  016704  000000G                 MOV     POOLPT,R4               ; GET ADDRESS OF BLOCK
000142  062767  000000G 000000G         ADD     #C$SIZE,POOLPT          ; UPDATE POOL FREE POINTER
000150  010410                          MOV     R4,(R0)                 ; LINK THE BLOCK INTO THE LIST
000152  005014                          CLR     (R4)                    ; MAKE IT THE LAST IN THE LIST
000154  005064  000000G                 CLR     C$FLAG(R4)              ; CLEAR THE FLAGS

000160  005264  000000G         40$:    INC     C$FLAG(R4)              ; INDICATE CONTEXT IN USE NOW
000164  010467  000000G                 MOV     R4,CURCTX               ; SET CURRENT CONTEXT ADDRESS
000170  005267  000000G                 INC     NUMUSR                  ; INDICATE ANOTHER USER
000174  012701  000000G                 MOV     #CONTEX,R1              ; GET ADDRESS OF LOW CORE CONTEXT
000200  012702  000000G                 MOV     #CTXSIZ,R2              ; GET LENGTH OF LOW CORE CONTEXT AREA
000204  105021                  45$:    CLRB    (R1)+                   ; CLEAR OUT CONTEXT AREA
000206                                  SOB     R2,45$                  ; IF NOT DONE, CLEAR NEXT BYTE
000212                                  SWSTK$  60$                     ;; ENTER SYSTEM STATE
000216  012701  000000G                 MOV     #W$SIZ,R1               ;; SET SIZE OF WINDOW BLOCK
000222                                  CALL    @AALOCB                 ;; ALLOCATE A BLOCK             ; RJK02
000226  103006                          BCC     50$                     ;; IF CC, SUCCESS
000230  017700  000000G                 MOV     @HEADR,R0               ;; ELSE, GET SAVED SP           ; RJK02
000234  011000                          MOV     (R0),R0                 ;; ... (@@)                     ;
000236  005260  000006                  INC     6(R0)                   ;; SET C-BIT
000242  000407                          BR      60$                     ;; AND RETURN
000244  010067  000000G         50$:    MOV     R0,TSKLN2               ;; POINT 2ND LUN WORD TO WINDOW BLOCK
000250  017720  000000G                 MOV     @TKTCB,(R0)+            ;; SET OUR TCB ADDRESS          ; RJK02
000254  010420                          MOV     R4,(R0)+                ;; ... CONTEXT AREA ADDRESS
000256  016510  000000G                 MOV     R$OPER(R5),(R0)         ;; ... OPERATE ROUTINE ADDRESS
000262                          60$:    RETURN
```

```
    52                          .SBTTL  MACRO CALLS AND LOCAL DATA
    53                  ;
    54                  ; MACRO LIBRARY CALLS
    55                  ;
    56                          .MCALL  MANDF$,PKTDF$,ALCCX$,MOCCX$,SAVRG,RESRG,CALLR
    57
    58 000000                   MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
    59 000000                   PKTDF$                          ; DEFINE I/O PACKET OFFSETS
    60
    61                  ;
    62                  ; CONTEXT AREA OFFSET DEFINITIONS
    63                  ;
    64 000000                   ALCCX$  LIST
                        ;
                        ; CONTEXT AREA OFFSET DEFINITIONS
                        ;
               000020           MAXID=16.                       ; MAXIMUM LENGTH OF USER ID
               000010           MAXPAS=8.                       ; MAXIMUM LENGTH OF PASSWORD
               000020           MAXACC=16.                      ; MAXIMUM LENGTH OF ACCOUNT
       000000                   .ASECT
               000000           .=0
                        ;
                        ; OFFSETS IN COMMON WITH READ INFORMATION MODULES
                        ;
       000000           A$FLG:  .BLKW   1                       ; FLAG WORD
       000002           A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
       000004           A$PAR:  .BLKW   1                       ; CURRENT PARAMETER TYPE
       000006           A$BUF:  .BLKW   1                       ; SAVED POINTER TO BUFFER
       000010           A$NUM:  .BLKB   1                       ; NUMBER OF BYTES IN ANB (FOR SET)
       000011           A$TYP:  .BLKB   1                       ; ALIAS FORMAT TYPE
       000012           A$ALI:  .BLKW   3                       ; ALIAS NAME, SPACE FILLED
       000020           A$SCO:  .BLKW   1                       ; SCOPE OF ALIAS
                        ;
                        ; OFFSETS UNIQUE TO CHANGE PARAMETERS MODULES
                        ;
       000022           A$DES:  .BLKW   3                       ; DESTINATION NAME, SPACE FILLED
       000030                   .BLKB   <1+MAXID>+<1+MAXPAS>+<1+MAXACC> ; ACCESS CONTROL STORAGE
       000103           A$LEN:                                  ; LENGTH OF CONTEXT AREA
       000000                   .PSECT
                        ;
                        ; FLAGS WORD BIT DEFINITIONS
                        ;
               000001           AF.INI=1                        ; SUCCESSFUL INITIALIZATION COMPLETED
               000002           AF.OVR=2                        ; CONTEXT AREA OVERFLOWED
    65 000000                   MOCCX$  LIST
                        ;
                        ; Context area offsets and flags
                        ;
       000103                   .ASECT
               000000           .=0
                        ;
                        ; THE FLAG VALUES, OPERATION STATUS INDICATORS AND COMMON PARAMETERS MUST
                        ; BE IDENTICAL AND AT THE SAME LOCATION IN BOTH MOCCX$ AND MORCX$.
                        ;
               000020           MXACC = 16.             ; Maximum account string length
               000010           MXPAS = 8.              ; Maximum password string length
               000020           MXUID = 16.             ; Maximum user id string length
```

```
          .TITLE  MANFRK - NETWORK MANAGEMENT ACP FORK PROCESSING
          .IDENT  /V05.00/
;
; COPYRIGHT (C) 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
;       NETWORK MANAGEMENT ACP FORK PROCESSING
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                        .SBTTL  MACRO CALLS AND LOCAL DATA
                                ;
                                ; MACRO CALLS
                                ;
                                        .MCALL  ALUN$,QIOW$S,CALLR,MANDF$,DIR$,QIOW$

000000                                  MANDF$                          ; DEFINE MANAGEMENT SYMBOLS
                                ;
                                ; LOCAL DEFINITIONS
                                ;
        000200                          CM$RSX=200                      ; DEFINE RSX SPECIFIC FUNCTION FLAG
        000454                          MXMES=300.                      ; LENGTH OF BUFFER
        177733                          IE.NLN=177733                   ; "NO FILE ON LUN" ERROR

                                        .IF DF R$$PRO ; PRO/DECNET

                                        NMFMES=20.                      ; LENGTH OF NICE BUFFER FOR NMFACP
                                        FILEUP=100264                   ; MP$FUP!<MO$PER*400>
                                                                        ; UPDATE FILE BY PERMANENT ACP (NMFACP)

                                        .ENDC   ; DF R$$PRO

                                ;
                                ; LOCAL MACRO DEFINITIONS
                                ;
                                        .MACRO  ERROR   IE,ME,TEXT
                                        $$$ = 0
                                        .IF NB <TEXT>
                                        .SAVE
                                        .PSECT  $$$STR,D,RO,LCL
                                        $$$ = .
                                        .ASCIZ  /TEXT/
                                        .RESTORE
                                        .ENDC
                                        .BYTE   IE,ME
                                        .WORD   $$$
                                        .ENDM   ERROR

                                ;
                                ; LOCAL STORAGE
                                ;
000000                          ALUN:   ALUN$   $TMLUN,NM               ; ASSIGN LUN
000010                          NMSTAT: .BLKW   2                       ; I/O STATUS BLOCK

                                        .IF DF R$$PRO ; PRO/DECNET

                                NMFBUF: .BLKB   NMFMES                  ; NICE BUFFER FOR NMFACP MESSAGES

                                        .ENDC   ; DF R$$PRO


                                ;
                                ; ERROR STRINGS
                                ;
000014                          ERRTBL:
000014                            ERROR IE.IDU, ME.OPE, <NM: driver not loaded>   ;ALUN$ ERROR
000020                            ERROR IE.HWR, ME.OPE, <NM: driver not resident> ;ALUN$ ERROR
```

```
                                                                   K 10
MANQIO      CREATED BY  MACRO  ON 29-JUN-85 AT 10:31       PAGE 1
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL   VALUE               REFERENCES

ALCHIN     000420 RG         #7-260
ALCHOP     000722 RG         #7-414
ALID       000520 R           7-263        7-267       #7-315
ALREIN     000426 RG         #7-264
ALREOP     000722 RG         #7-415
ALUN       000000 R          #5-92         6-156
ARID       000652 R           7-274       #7-383
ARREIN     000434 RG         #7-271
ARREOP     000722 RG         #7-416
A.LULU   = 000002             5-92
CICHIN     000442 RG         #7-278
CICHOP     000722 RG         #7-417
CIID       000520 R           7-281        7-285        7-289       #7-316
CIREIN     000450 RG         #7-282
CIREOP     000722 RG         #7-418
CIZEIN     000456 RG         #7-286
CIZEOP     000722 RG         #7-419
CM$RSX   = 000200            #5-61         7-262        7-266        7-310        7-314        7-378        7-382        7-394        7-398
                              7-402
ERRTBL     000014 R          #5-105        6-172
IE.ABO   = ******   GX        5-112
IE.DAO   = ******   GX        5-109
IE.HWR   = ******   GX        5-107
IE.IDU   = ******   GX        5-106
IE.NDR   = ******   GX        5-114
IE.NFW   = ******   GX        5-113
IE.NLN   = 177733            #5-63         5-108
IE.NST   = ******   GX        5-111
IE.OFL   = ******   GX        5-115
IE.PRI   = ******   GX        5-110
INIQIO     000070 R          #6-155        7-408
LICHIN     000464 RG         #7-293
LICHOP     000722 RG         #7-420
LIID       000520 R           7-296        7-300        7-304       #7-317
LIREIN     000472 RG         #7-297
LIREOP     000722 RG         #7-421
LIZEIN     000500 RG         #7-301
LIZEOP     000722 RG         #7-422
LOCHIN     000532 RG         #7-326
LOCHOP     000722 RG         #7-423
LOID       000544 R           7-329       #7-333
LOREIN     000540 RG         #7-330
LOREOP     000722 RG         #7-424
MD.NON     177777             6-170        6-171
ME$ALI     000010             7-261        7-265
ME$ARE     000005             7-272
ME$CIR     000003             7-279        7-283        7-287
ME$LIN     000001             7-294        7-298        7-302
ME$LOG     000002             7-327        7-331
ME$MOD     000004             7-339        7-343        7-347
ME$NOD     000000             7-356        7-360        7-364
ME$OBJ     000007             7-377        7-381
```

```
                                                                   L 10
MANQIO      CREATED BY  MACRO  ON 29-JUN-85 AT 10:31       PAGE 2
SYMBOL CROSS REFERENCE                                     CREF    04.00
```

```
                                                                                        K 11
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31  Page 9-2
Symbol table

A$$CHK= 000000            MC$CAP  001440            MC$SYS  000024            MD$FUL  000000            ME.CST  177765
A$$CPS= 000000            MC$CCL  001445            MC$SYZ  000000            MD$HAL  000001            ME.DIS  177755
A$$PRI= 000000            MC$CDC  002045            MC$TBR  001140            MD$HEL  000002            ME.DON  177600
A$$TRP= 000000            MC$CDP  001441            MC$TBS  001141            MD$KCP  000013            ME.FCO  177762
A.ACC   000020            MC$CIF  001465            MC$TDR  002053            MD$KDP  000034            ME.FIO  177756
A.LNK   000000            MC$CLD  001464            MC$TIM  000000            MD$KDZ  000036            ME.FOP  177763
A.NAM   000002            MC$COU  100000            MC$TYP  007777            MD$KL   000040            ME.FOR  177776
A.PUD   000010            MC$CTL  001454            MC$UBU  002052            MD$KMX  000054            ME.FUN  177777
A.REM   000012            MC$CTR  001452            MC$UFD  002047            MD$KMY  000052            ME.GRO  177745
A.UCB   000010            MC$CTS  001453            MC$UMR  001132            MD$LEN  000050            ME.HAR  177750
BIAS  = 000002            MC$DOV  002050            MC$UMS  001133            MD$PCL  000011            ME.IID  177767
CURIOP= ****** GX         MC$LBR  001750            MC$VAX  000003            MD$QNA  000005            ME.LCO  177766
C$$CKP= 000000            MC$LBS  001751            MC$WID  060000            MD$UNA  000001            ME.LPR  177757
C$$ORE= 000400            MC$LDI  001774            MC$WIH  040000            MD$WIT  000002            ME.MPR  177773
C$$RSH= 177564            MC$LDO  001775            MC$WIL  020000            MD.ABO  000017            ME.MVE  177771
D$$BUG= 177514            MC$LDR  001762            MC$W08  020000            MD.ACC  000010            ME.OPE  177747
D$$ISK= 000000            MC$LDS  001763            MC$W16  040000            MD.AOB  000016            ME.PLO  177751
D$$L11= 000001            MC$LLB  002021            MC$W32  060000            MD.BOB  000011            ME.PMI  177743
D$$YNC= 000000            MC$LLP  002115            MC$XBR  001750            MD.DIA  000007            ME.PNA  177752
D$$YNM= 000000            MC$LLR  002007            MC$XBS  001751            MD.DOB  000015            ME.PRI  177775
E$$XPR= 000000            MC$LOD  000001            MC$XCJ  000322            MD.DUM  000002            ME.PTY  177772
F$$LVL= 000001            MC$LRB  002020            MC$XCR  002260            MD.FAI  000014            ME.PVA  177760
G$$TPP= 000000            MC$LRP  002114            MC$XCS  002261            MD.FNA  000001            ME.RES  177761
G$$TSS= 000000            MC$LRR  002006            MC$XDR  001762            MD.FOB  000006            ME.ROO  177754
G$$TTK= 000000            MC$LSI  002032            MC$XDS  001763            MD.LOA  000001            ME.SIZ  177774
G$$WRD= 000000            MC$LST  002033            MC$XFR  002272            MD.LSH  000020            ME.SYS  177746
I$$RAR= 000000            MC$LUP  006072            MC$XFS  002273            MD.NOB  000012            ME.UCO  177770
I$$RDN= 000000            MC$MAP  010000            MC$XLJ  000323            MD.NON  177777            MF$ACT  177776
I.PRM = ****** GX         MC$MBL  001764            MC$XLR  002330            MD.PER  000000            MF$ADD  000000
KISAR6= ****** GX         MC$MBX  004420            MC$XMA  000310            MD.RES  000004            MF$ADJ  177774
K$$CNT= 177546            MC$MBY  001752            MC$XMC  002305            MD.ROB  000005            MF$ALL  177775
K$$CSR= 177546            MC$MPX  004406            MC$XMS  002304            MD.RSH  000013            MF$BYE  000300
K$$LDC= 000000            MC$NAP  001604            MC$XNR  002332            MD.SDU  000005            MF$CHA  000023
K$$TPS= 000074            MC$NBR  001130            MC$XRC  002316            MD.SLO  000003            MF$DUM  000020
LC$EXT  000000            MC$NBS  001131            MC$XRR  002331            MD.SNA  000000            MF$EVT  000001
LC$INT  000001            MC$NCR  001154            MC$XRS  002342            MD.TLO  000004            MF$FUP  000264
LD$LP = 000000            MC$NCS  001155            MC$020  000002            MD.UNA  000002            MF$KNO  177777
L$$ASG= 000000            MC$NML  001274            MD$CI   000007            MD.UNR  000003            MF$LOA  000017
L$$DRV= 000000            MC$NMR  001142            MD$CNA  000003            MD.UOB  000007            MF$LOO  177775
L$$P11= 000001            MC$NMS  001143            MD$COU  000001            MD.VOL  000006            MF$REA  000024
L$$11R= 000000            MC$NNO  001606            MD$DA   000010            ME$ALI  000010            MF$SIG  177773
MAPADD= ****** GX         MC$NNU  001605            MD$DL   000004            ME$AL2  000003            MF$SPF  000302
MAPALI= ****** GX         MC$NOP  001607            MD$DLV  000020            ME$ARE  000005            MF$SYS  000026
MAPREM= ****** GX         MC$NOR  000000            MD$DMC  000014            ME$CIR  000003            MF$TES  000022
MB$DIS  000001            MC$NPF  001616            MD$DMF  000046            ME$EXA  000000            MF$TRA  000301
MB$ENA  000000            MC$NPR  001630            MD$DMP  000022            ME$EXE  000200            MF$TRI  000021
MB$FUL  000002            MC$NRE  001200            MD$DMR  000050            ME$LIN  000001            MF$ZER  000025
MB$MIX  000002            MC$NRT  001166            MD$DMV  000042            ME$LOG  000002            ML$ALL  100000
MB$ONE  000001            MC$NVR  001642            MD$DN   000016            ME$MOD  000004            ML$CLS  000000
MB$RXO  000001            MC$P08  000000            MD$DP   000000            ME$NOD  000000            ML$CON  000001
MB$TXO  000000            MC$P11  000001            MD$DPV  000044            ME$NON  177777            ML$EXT  000001
MB$ZER  000000            MC$RFL  002046            MD$DQ   000006            ME$OBJ  000007            ML$FIL  000002
MC$BAB  006073            MC$SBU  002051            MD$DTE  000024            ME$OB2  000004            ML$FIR  000000
MC$BID  001765            MC$SFL  002044            MD$DU   000002            ME$PRO  000005            ML$INT  000000
MC$BMC  001767            MC$SYC  000012            MD$DUP  000012            ME$SYS  000006            ML$KNO  140000
MC$BSC  001766            MC$SYL  000036            MD$DV   000026            ME.BLO  177744            ML$MON  000003
MC$CAC  001442            MC$SYR  000050            MD$DZ   000030            ME.CON  177753            ML$SYS  000002
```

```
                                                                                        L 11
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31  Page 9-3
Symbol table
```

```
                                                       K 12
MANVCD - VOLATILE MANAGEMENT AC MACRO V05.03b Saturday 29-Jun-85 10:31  Page 7-3
Symbol table

MP$XPL  000244         MS$TOP  000003         MT$TER  000001         M$$CRX= 000000         X$$DBT= 000000
MP$XPT  000240         MS$TRI  000005         MT$TRI  000002         M$$FCS= 000000         ZF.COU= 001000
MP$XXX  177777         MS$UNR  000005         MT$TYP  007777         M$$MGE= 000000         ZF.DDM= 000001
MP$5ST  004406         MS$VMS  000004         MT$USD  000000         M$$NET= 000000         ZF.DIA= 004000
MS$ACT  000001         MS.DON  177600         MT$X25  000003         M$$OVR= 000000         ZF.DLC= 000002
MS$ADU  000010         MS.MOR  000002         MU$INC  000001         NM.CLN  003400         ZF.DVP= 100000
MS$ALD  000007         MS.PAR  000003         MU$OUT  000002         NM.INI  002400         ZF.INI= 040000
MS$ASE  000006         MS.SUC  000001         MU$PER  000000         NM.OPR  003000         ZF.KMX= 000020
MS$ATR  000011         MT$AR4  000003         MV$III  000000         NM.VR2  002401         ZF.LLC= 000004
MS$AUT  000000         MT$ASC  000100         MV$IV   000001         NM.VR3  002402         ZF.LMC= 000100
MS$CLE  000003         MT$BIL  000001         MX$ACT  000020         N$$ACC= 000001         ZF.MAN= 020000
MS$DED  000004         MT$BYS  000011         MX$CIR  000020         N$$BUF= 000001         ZF.MFL= 000010
MS$DIE  000003         MT$CI   000007         MX$CLN  000020         N$$LDV= 000001         ZF.MTM= 000400
MS$DUM  000004         MT$CLE  000077         MX$CNM  000006         N$$MCP= 000001         ZF.MUX= 000040
MS$FAI  000013         MT$COD  000200         MX$CON  000006         N$$MLL= 000001         ZF.PSE= 002000
MS$GLO  000000         MT$CON  000001         MX$DAC  000020         N$$MOV= 000010         ZF.SLI= 010000
MS$HOL  000002         MT$DMC  000004         MX$DTE  000020         N$$NCT= 000001         ZF.TIM= 000200
MS$INA  000002         MT$ETH  000006         MX$FIL  000034         N$$PEM= 000001         ZF.X3P= 000000
MS$LOA  000003         MT$HEX  000040         MX$LGC  000777         PDVID = ****** GX      ZS.ASN= 100000
MS$LOO  000002         MT$LPB  000005         MX$LIN  000020         PDVTA = ****** GX      ZS.BSY= 140000
MS$OFF  000001         MT$MAX  000037         MX$LON  000006         POOLPT= ****** GX      Z.AVL   000014
MS$ON   000000         MT$MUL  000100         MX$NMS  000454         P$$P45= 000000         Z.DAT   000016
MS$PRO  000006         MT$NLE  000017         MX$NOD  000006         P$$WRD= 000000         Z.DSP   000000
MS$REA  000004         MT$NON  000001         MX$OBJ  000006         Q$$OPT= 000010         Z.FLG   000010
MS$REF  000001         MT$NR4  000005         MX$OWN  000040         R$$DER= 000000         Z.LEN = 000016
MS$RES  000003         MT$NTY  000060         MX$PAR  000036         R$$K11= 000001         Z.LLN   000006
MS$RST  000001         MT$OCT  000060         MX$PAS  000010         R$$SND= 000000         Z.MAP   000020
MS$RSX  000002         MT$PHA  000002         MX$RAC  000047         R$$11M= 000000         Z.NAM   000004
MS$RT   000005         MT$POI  000000         MX$RID  000047         SF$ANM  000001         Z.PCB   000012
MS$SER  000002         MT$QP2  000010         MX$RPS  000047         SF$DNM  000002         Z.SCH   000007
MS$SHU  000002         MT$ROU  000000         MX$SID  000040         S$$WRG= 000000         $HEADR= ****** GX
MS$SRV  000007         MT$RO4  000004         MX$SNK  000377         S$$YSZ= 007600         $MBLUN= ****** GX
MS$STA  000000         MT$SEC  000000         MX$TYP  000077         T$$KMG= 000000         $$$SYS= 004374
MS$SYN  000012         MT$SGD  000020         MX$UID  000020         T$$MIN= 000000         $$$SYX= 000000
MS$TER  000001         MT$SYS  000002         M$$CRB= 000124         V$$CTR= 001000

. ABS.  000020    000   (RW,I,GBL,ABS,OVR)
        000154    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 203
Work  file writes: 137
Size of work file: 26784 Words  ( 105 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:23.13
SY:MANVCD.V2,[135,134]MANVCD/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANVCD
```

```
                                                           L 12
MANVCD      CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
                                                                  K 13
MANVIB     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF   04.00
SYMBOL  VALUE            REFERENCES

ALCHIN  = ******  GX      6-101     6-102
ALCHOP  = ******  GX      6-101     6-102
ALREIN  = ******  GX      6-103
ALREOP  = ******  GX      6-103
ARREIN  = ******  GX      6-95
ARREOP  = ******  GX      6-95
CICHIN  = ******  GX      6-108     6-109
CICHOP  = ******  GX      6-108     6-109
CIREIN  = ******  GX      6-111
CIREOP  = ******  GX      6-111
CIZEIN  = ******  GX      6-110
CIZEOP  = ******  GX      6-110
CM$RSX  = 000200         #5-58      6-101     6-102     6-103     6-149     6-150     6-151     6-156     6-157
                          6-159     6-164     6-165     6-170     6-171
FTABLE    000000 RG      #6-91
LICHIN  = ******  GX      6-115     6-116
LICHOP  = ******  GX      6-115     6-116
LIREIN  = ******  GX      6-118
LIREOP  = ******  GX      6-118
LIZEIN  = ******  GX      6-117
LIZEOP  = ******  GX      6-117
LOCHIN  = ******  GX      6-122     6-123
LOCHOP  = ******  GX      6-122     6-123
LOGOUT  = ******  GX      6-185
LOREIN  = ******  GX      6-124
LOREOP  = ******  GX      6-124
ME$ALI    000010          6-101     6-102     6-103
ME$ARE    000005          6-95
ME$CIR    000003          6-108     6-109     6-110     6-111
ME$LIN    000001          6-115     6-116     6-117     6-118
ME$LOG    000002          6-122     6-123     6-124
ME$MOD    000004          6-131     6-132     6-133     6-134
ME$NOD    000000          6-141     6-142     6-143     6-144
ME$OBJ    000007          6-149     6-150     6-151
ME$PRO    000005          6-156     6-157     6-159
ME$SYS    000006          6-164     6-165     6-170     6-171
MF$BYE    000300          6-185
MF$CHA    000023          6-101     6-102     6-108     6-109     6-115     6-116     6-122     6-123     6-131
                          6-132     6-141     6-142     6-149     6-150     6-156     6-157     6-164     6-165
MF$REA    000024          6-95      6-103     6-111     6-118     6-124     6-134     6-144     6-151     6-159
                          6-171
MF$TRA    000301          6-179     6-180     6-181     6-182
MF$ZER    000025          6-110     6-117     6-133     6-143     6-170
MOCHIN  = ******  GX      6-131     6-132
MOCHOP  = ******  GX      6-131     6-132
MOREIN  = ******  GX      6-134
MOREOP  = ******  GX      6-133     6-134
MOZEIN  = ******  GX      6-133
MO$ADD    000002          6-181
MO$ALA    000004          6-182
MO$ALI    000000          6-179
MO$CLE    000100          6-102     6-109     6-116     6-123     6-132     6-142     6-150     6-157     6-165
```

```
                                                                  L 13
MANVIB     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00
```

Table of contents

```
     5-   53    MACRO CALLS AND LOCAL DATA
     6-   64    ALCHIN - ALIAS CHANGE PARAMETERS INITIALIZE
     7-  167    CHKDES - CHECK DESTINATION FORMAT
     8-  254    CHKACC - CHECK ACCESS CONTROL FIELDS
```

```
**FILE**ID**ALCHOP
```

```
  AAAAAA     LL           CCCCCCCC   HH      HH     OOOOOO     PPPPPPPP
  AAAAAA     LL           CCCCCCCC   HH      HH     OOOOOO     PPPPPPPP
AA      AA   LL           CC           HH      HH   OO      OO   PP      PP
AA      AA   LL           CC           HH      HH   OO      OO   PP      PP
AA      AA   LL           CC           HH      HH   OO      OO   PP      PP
AA      AA   LL           CC           HH      HH   OO      OO   PP      PP
AA      AA   LL           CC           HHHHHHHHHH   OO      OO   PPPPPPPP
AA      AA   LL           CC           HHHHHHHHHH   OO      OO   PPPPPPPP
AAAAAAAAAA   LL           CC           HH      HH   OO      OO   PP
AAAAAAAAAA   LL           CC           HH      HH   OO      OO   PP
AA      AA   LL           CC           HH      HH   OO      OO   PP                ....
AA      AA   LL           CC           HH      HH   OO      OO   PP                ....
AA      AA   LLLLLLLLLL     CCCCCCCC   HH      HH     OOOOOO     PP                ....
AA      AA   LLLLLLLLLL     CCCCCCCC   HH      HH     OOOOOO     PP                ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
ALCHOP - ALIAS CHANGE PARAMETER MACRO V05.03b Tuesday 10-Sep-85 11:45
Table of contents
```

```
                                                                       K 16
ALCHOP - ALIAS CHANGE PARAMETER MACRO V05.03b Tuesday 10-Sep-85 11:45  Page 10-2
Symbol table

MD.NOB   000012       ME.UCO   177770       MO$PER   000200       MP$CPL   000232       MP$LCT   000156
MD.NON   177777       MF$ACT   177776       MO$PRO   000002       MP$CPT   000226       MP$LHL   000232
MD.PER   000000       MF$ADD   000000       MO$PRS   000001       MP$CPU   000161       MP$LLE   000227
MD.RES   000004       MF$ADJ   177774       MO$REA   000200       MP$CSZ   001755       MP$LLO   004432
MD.ROB   000005       MF$ALL   177775       MO$SET   000000       MP$CUS   002127       MP$LMB   002152
MD.RSH   000013       MF$BYE   000300       MO$STA   000020       MP$CVA   000540       MP$LMX   002200
MD.SDU   000005       MF$CHA   000023       MO$SUM   000000       MP$DAL   002570       MP$LNA   000144
MD.SLO   000003       MF$DUM   000020       MO$TSK   000000       MP$DCO=  001464       MP$LNO   000233
MD.SNA   000000       MF$EVT   000001       MO$VOL   000000       MP$DDT   002177       MP$LOA   000170
MD.TLO   000004       MF$FUP   000264       MO$ZER   000000       MP$DEL   001131       MP$LOG   004514
MD.UNA   000002       MF$KNO   177777       MP$ACB   000012       MP$DES   000156       MP$LOO   000620
MD.UNR   000003       MF$LOA   000017       MP$ACC   000514       MP$DEV   002114       MP$LTM   001613
MD.UOB   000007       MF$LOO   177775       MP$ACT   004526       MP$DFA   001320       MP$LTY   004533
MD.VOL   000006       MF$REA   000024       MP$ADD   000766       MP$DHO=  001465       MP$LWI   000230
ME$ALI   000010       MF$SIG   177773       MP$ADJ   001440       MP$DIA   000173       MP$MAC   004420
ME$AL2   000003       MF$SPF   000302       MP$ADP   006325       MP$DLB   002571       MP$MAD   001630
ME$ARE   000005       MF$SYS   000026       MP$ADS   004406       MP$DLG   004521       MP$MAP   006323
ME$CIR   000003       MF$TES   000022       MP$ALB   000036       MP$DLI=  001466       MP$MAR   001635
ME$EXA   000000       MF$TRA   000301       MP$AMC   001640       MP$DLT   002200       MP$MAV   006321
ME$EXE   000200       MF$TRI   000021       MP$AMH   001641       MP$DRO   001441       MP$MBN   001636
ME$LIN   000001       MF$ZER   000025       MP$ANB   004420       MP$DST   000454       MP$MBR   001637
ME$LOG   000002       ML$ALL   100000       MP$ASB   000024       MP$DTE   002140       MP$MBU   001642
ME$MOD   000004       ML$CLS   000000       MP$ASC   004432       MP$DTY=  001452       MP$MCB   000156
ME$NOD   000000       ML$CON   000001       MP$AUS   000512       MP$DUA   000207       MP$MCO   001632
ME$NON   177777       ML$EXT   000001       MP$BBT   002165       MP$DUC   000210       MP$MDE   004421
ME$OBJ   000007       ML$FIL   000002       MP$BDF   002164       MP$DUM   000202       MP$MHO   001633
ME$OB2   000004       ML$FIR   000000       MP$BFQ   002121       MP$DUP   002127       MP$MLB   000202
ME$PRO   000005       ML$INT   000000       MP$BLK   001616       MP$DVC   002114       MP$MLK   001306
ME$SYS   000006       ML$KNO   140000       MP$BLO   001452       MP$DWE   001321       MP$MLN   001631
ME.BLO   177744       ML$MON   000003       MP$BMX   002176       MP$ELT   000157       MP$MLP   006333
ME.CON   177753       ML$SYS   000002       MP$BNP   006330       MP$ETY   001605       MP$MRB   002171
ME.CST   177765       ML$TOP   000001       MP$BRT   001620       MP$EVE   000311       MP$MRO   001605
ME.DIS   177755       MN$UNL   000377       MP$BSA   002176       MP$FNC   001752       MP$MRP   006322
ME.DON   177600       MO$ACC   000200       MP$BSD   002203       MP$GDT   002222       MP$MRT   002153
ME.FCO   177762       MO$ADD   000002       MP$BSI   002200       MP$GNM   002223       MP$MRV   006320
ME.FIO   177756       MO$ALA   000004       MP$BSP   006331       MP$GRO   002115       MP$MSB   000170
ME.FOP   177763       MO$ALI   000000       MP$BUF   000170       MP$GRP   000541       MP$MVE   000145
ME.FOR   177776       MO$CHA   000040       MP$BUP   006324       MP$GTY   002224       MP$MVI   001634
ME.FUN   177777       MO$CIR   000003       MP$BUS   001643       MP$HAD   001757       MP$MVR   001751
ME.GRO   177745       MO$CLE   000100       MP$CAC   001750       MP$HBT   002142       MP$MWN   002154
ME.HAR   177750       MO$COU   000060       MP$CAS   001762       MP$HDD   000162       MP$MXB   002172
ME.IID   177767       MO$DAC   000000       MP$CAT   002210       MP$HTM   001612       MP$MXC   000466
ME.LCO   177766       MO$DEF   000000       MP$CCS   004406       MP$HWA   002210       MP$MXR   001630
ME.LPR   177757       MO$DPR   000000       MP$CHN   002141       MP$IAT   001322       MP$MXW   002143
ME.MPR   177773       MO$ENT   000017       MP$CIR   000144       MP$IDE   000144       MP$NAA   006335
ME.MVE   177771       MO$EVE   000100       MP$CLK   002131       MP$IDP   006327       MP$NAC   001130
ME.OPE   177747       MO$INF   000160       MP$CLN   002126       MP$IHO   000215       MP$NAP   006326
ME.PLO   177751       MO$INS   000002       MP$CLT   002211       MP$INA   002177       MP$NCT   000240
ME.PMI   177743       MO$LIN   000001       MP$CMB   002142       MP$IND   002204       MP$NET   002114
ME.PNA   177752       MO$LOG   000001       MP$CMK   000537       MP$INI   002201       MP$NLI   000765
ME.PRI   177775       MO$MIR   000031       MP$CMX   002153       MP$ITI   000776       MP$NNA   000764
ME.PTY   177772       MO$NAM   000001       MP$CND   000310       MP$LAA   000231       MP$NOD   000500
ME.PVA   177760       MO$NIC   000023       MP$CNU   001753       MP$LAN   000234       MP$NRA   006334
ME.RES   177761       MO$NOD   000000       MP$COB   000311       MP$LAR   000202       MP$NRB   000214
ME.ROO   177754       MO$OFF   000001       MP$CON   002126       MP$LAS   000012       MP$NSA   001617
ME.SIZ   177774       MO$ON    000000       MP$COS   001604       MP$LBS   000203       MP$NTI   002141
ME.SYS   177746       MO$OPT   000001       MP$CPF   000230       MP$LCO   000226       MP$NUM   001642
```

```
                                                                       L 16
ALCHOP - ALIAS CHANGE PARAMETER MACRO V05.03b Tuesday 10-Sep-85 11:45  Page 10-3
Symbol table
```

```
    870 001162  012700  000166'                     MOV     #WORKSP,R0              ; Get address of output event string
    871 001166  012701  000073'                     MOV     #PARERR,R1              ; Copy address of $EDMSG instructions
    872 001172  010302                              MOV     R3,R2                   ; Copy address of $EDMSG arguments
    873 001174                                      CALL    $EDMSG                  ; Format the output string
    874 001200  105010                              CLRB    (R0)                    ; Make it an ASCIZ string
    875 001202  012702  000166'                     MOV     #WORKSP,R2              ; Recover error-specific string address
    876 001206                                      CALL    LOGEVT                  ; Log the event
    877 001212  000261                              SEC                             ; Set C-bit to indicate load error
    878 001214  000410                              BR      30$                     ; Branch to common code
    879
    880 001216  010267  000124'          20$:       MOV     R2,PCBADR               ; Save PCB address
    881 001222  016267  000014  000152'             MOV     P.REL(R2),BLOCK         ; Copy partition mapping bias
    882 001230  016267  000016  000154'             MOV     P.SIZE(R2),BLOCK+2      ; Copy partition size (32-word blocks)
    883
    884 001236  012626                   30$:       MOV     (SP)+,(SP)+             ; Deallocate work area but save C-bit
    885 001240  012603                              MOV     (SP)+,R3                ; Retrieve address of delimiter
    886 001242  112613                              MOVB    (SP)+,(R3)              ; Replace it with original data
    887 001244                                      RETURN                          ; Exit
    888
    889                                             .IFF                            ; If no PLAS support ...
```

```
                                                                   L  2
MLD11S      CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 7
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE              REFERENCES

STDDM     000622 RG        #8-245       12-674
SV$DUM  = 000001           #6-83
SV$LOA  = 000000           #6-83
S.COST    000001           #6-85
S.FLG     000000           #6-85
S.LEN     000004           #6-85
S.NMST    000002           #6-85
S.OWNR    000003           #6-85
S.TSEF  = 000002            8-203
TKTCB     000602 RG        #8-237       9-358        12-668
TMREFL  = 000003           #7-115       8-202        8-203
T.RCVL    000012            9-359       12-669
WAIT      000140 R         #8-203       9-341
WORKSP    000166 R         #8-215       16-870       16-875
X$$MCB  = ******            6-84        6-84
ZF.COU  = 001000           #6-84
ZF.DDM  = 000001           #6-84
ZF.DIA  = 004000           #6-84
ZF.DLC  = 000002           #6-84
ZF.DVP  = 100000           #6-84
ZF.INI  = 040000           #6-84
ZF.KMX  = 000020           #6-84
ZF.LLC  = 000004           #6-84
ZF.LMC  = 000100           #6-84
ZF.MAN  = 020000           #6-84
ZF.MFL  = 000010           #6-84
ZF.MTM  = 000400           #6-84
ZF.MUX  = 000040           #6-84
ZF.PSE  = 002000           #6-84
ZF.SLI  = 010000           #6-84
ZF.TIM  = 000200           #6-84
ZF.X3P  = 000000           #6-84
ZS.ASN  = 100000           #6-84
ZS.BSY  = 140000           #6-84
Z.AVL     000014           #6-84
Z.DAT     000016           #6-84
Z.DSP     000000           #6-84        6-84
Z.FLG     000010           #6-84
Z.LEN   = 000016           #6-84
Z.LLN     000006           #6-84
Z.MAP     000020           #6-84
Z.NAM     000004           #6-84        11-604
Z.PCB     000012           #6-84
Z.SCH     000007           #6-84
$BLXIO  = ******    GX      8-230
$CAT5   = ******    GX      16-848      16-852
$CCBRT  = ******    GX      8-238
$EDMSG  = ******    GX      11-609      16-873
$LDBGT  = ******    GX      8-239
$LDBRT  = ******    GX      8-240
$NTUIC  = ******    GX      8-234
$PARHD  = ******    GX      8-235
```

```
                                                                   M  2
MLD11S      CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 8
```

```
001604  016302  000004                   MOV     C.NSP(R3),R2            ;; GET ADDRESS OF CONTEXT AREA FROM CCB
001610  010362  000012                   MOV     R3,C$CCB(R2)            ;; PASS BACK CCB ADDRESS
001614  052762  000002  000002           BIS     #CF$FDN,C$FLAG(R2)      ;; FLAG CONTEXT AS BEING RESUMEABLE
001622  000472                           BR      45$                     ;; AND GO DO IT!
                                         .ENDC   ;DF MN$VOL

                                 ;
                                 ; SCAN THE RECEIVE QUEUE LOOKING FOR PACKETS NEEDING PROCESSING
                                 ;
001624  010005                   13$:    MOV     R0,R5                   ;; KEEP A COPY OF TCB ADDRESS
001626  017760  176162  000000G          MOV     @COPT,T.UCB(R0)         ;; DEFAULT OUR TI: TO CO:          ; RJK02
001634  062700  000000G                  ADD     #T.RCVL,R0              ;; POINT TO THE I/O PACKET LISTHEAD
001640  011003                   15$:    MOV     (R0),R3                 ;; GET NEXT PACKET ADDRESS
001642  001406                           BEQ     17$                     ;; IF EQ, NONE REMAINING
001644  032763  000001  000004           BIT     #1,I.TCB(R3)            ;; HAS THIS PACKET BEEN SEEN ALREADY ?
001652  001414                           BEQ     30$                     ;; IF EQ, NO - USE IT
001654  010300                           MOV     R3,R0                   ;; ELSE, CONTINUE SEACH FROM HERE
001656  000770                           BR      15$                     ;; ...
                                 ;
                                 ; NO NEW I/O PACKETS, LOOK FOR COMPLETED FORKS
                                 ;
001660  005767  176340           17$:    TST     FDNCNT                  ;; ARE THERE ANY FORKS TO BE RESUMED ?
001664  001051                           BNE     45$                     ;; IF NE, YES - GO FINISH PROCESSING THME
                                 ;
                                 ; NO COMPLETED FORKS, IF THERE ARE CURRENT USERS, STOP THE TASK, ELSE EXIT
                                 ;
001666  005767  176326                   TST     NUMUSR                  ;; ELSE, ARE THERE ANY CURRENT USERS ?
001672  001402                           BEQ     20$                     ;; IF EQ, NO - EXIT
001674                                   CALLR   @STPCT                  ;; ELSE, STOP TASK FOR NOW         ; RJK02
001700                           20$:    CALLR   @DREXT                  ;; EXIT TASK                       ; RJK02
                                 ;
                                 ; FOUND AN I/O PACKET. FLAG IT AS BEING PROCESSED
                                 ;
001704  010367  176300           30$:    MOV     R3,CURIOP               ;; SAVE I/O PACKET ADDRESS
001710  016300  000004                   MOV     I.TCB(R3),R0            ;; GET USER'S TCB ADDRESS
001714  005263  000004                   INC     I.TCB(R3)               ;; FLAG THIS PACKET AS BEING PROCESSED
                                         .IF DF  R$$MPL
                                         SAVRG   <R0,R3>                 ;; SAVE TCB AND I/O PACKET
                                         CALL    @TICLR                  ;; CLEAR CURRENT TI:               ; RJK02
                                         RESRG   <R3,R0>                 ;; RESTORE I/O PACKET AND TCB
                                         .ENDC
001720  016065  000000G 000000G          MOV     T.UCB(R0),T.UCB(R5)     ;; ... AND SET OUR TI: TO HIS
                                 ;
                                 ; COPY THE TASK'S 2ND LUN WORD INTO LOCAL STORAGE
                                 ;
                                         .IF DF  R$$MPL
                                         MOV     T.PCB(R0),R5            ;; GET PARTITION CONTROL BLOCK ADDRESS
                                         SAVMAP                          ;; SAVE CURRENT MAPPING
                                         MAP     P.REL(R5)               ;; MAP TO THE TASK HEADER
                                         .IFTF   ;DF R$$MPL
001726  017367  000006  176266           MOV     @I.LN2(R3),TSKLN2       ;; COPY THE TASK'S SECOND LUN WORD
001734  042767  000001  176260           BIC     #1,TSKLN2               ;; CLEAR LUN INTERLOCK BIT
                                         .IFT    ;DF R$$MPL
                                         RESMAP                          ;; RESTORE PREVIOUS MAPPING
                                         .ENDC   ;DF R$$MPL
                                 ;
                                 ; COPY ID STRING AND INPUT DATA FOR NM.INI REQUESTS
                                 ;
```

```
                                                                                      L 4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-5
Symbol table

MX$NMS  000454              O.ESB   000012 G            R$R2  = 000006 G            UA.PUT= 000040              $LDBRT= ****** GX
MX$NOD  000006              O.LGTH= 000034              R$R3  = 000010 G            UA.SPE= 000020              $LGCON= ****** GX
MX$OBJ  000006              O.LNK   000000 G            R$R4  = 000012 G            UA.TRA= 002000              $LGFNB= ****** GX
MX$OWN  000040              O.MCRL  000002 G            R$R5  = 000014 G            UA.TRN= 001000              $LGMON= ****** GX
MX$PAR  000036              O.PTCB  000004 G            R$SIZE  000010 G            UA.TYP= 000010              $LGSTT= ****** GX
MX$PAS  000010              O.STAT  000014 G            R$$DER= 000000              UPDCTX= ****** GX           $LGUIC= ****** GX
MX$RAC  000047              PC.ALF= 000004              R$$K11= 000001              U.CW2 = ****** GX           $LLCRS= ****** GX
MX$RID  000047              PC.ALM= 001000              R$$SND= 000000              U2.PRV= ****** GX           $LLCTA= ****** GX
MX$RPS  000047              PC.HIH= 000001              R$$11M= 000000              V$$CTR= 001000              $MOULS= ****** GX
MX$SID  000040              PC.LOW= 000002              SB.ACK= 177776              WBUFF = ****** GX           $NMCLH= ****** GX
MX$SNK  000377              PC.NRM= 000400              SB.BOF= 177773              W$ACP   000000 G            $NMCL2= ****** GX
MX$TYP  000077              PC.XAF= 000010              SB.DIS= 177771              W$CTX   000002 G            $NMCRS= ****** GX
MX$UID  000020              PC.XIT= 000200              SB.ENQ= 177774              W$OPR   000004 G            $NMLST= ****** GX
M$$CRB= 000124              PDVID   000134RG            SB.KIL= 177777              W$SIZ   000006 G            $NTLPT= ****** GX
M$$CRX= 000000              PDVNM   000136RG            SB.NAK= 177775              XAVL    000172RG            $OBJHD= ****** GX
M$$FCS= 000000              PDVTA   000140RG            SB.TMO= 177772              XBIAS   000174RG            $PDVID= ****** GX
M$$MGE= 000000              PF.ALL= 177777              SDBAF   000156RG            X$$DBT= 000000              $PDVNM= ****** GX
M$$NET= 000000              PF.INS= 000040              SDBCT   000160RG            ZTIME   000176RG            $PDVTA= ****** GX
M$$OVR= 000000              PF.LOG= 000100              SDBNM   000162RG            ZTIM2   000200RG            $PRUNM  000204RG
NDRERR  002250R             PF.REQ= 000200              SF$ANM  000001              $ALOCB= ****** GX           $PSIPT= ****** GX
NLNERR  002256R             POOLPT  000234RG            SF$DNM  000002              $BLXIO= ****** GX           $RDBAF= ****** GX
NMCLH   000120RG            POOLSZ  000236RG            SLTMA   000164RG            $CALLX= ****** GX           $RDBCT= ****** GX
NMCL2   000122RG            PRIERR  002264R             SLTNM   000166RG            $CCBAF= ****** GX           $RDBNM= ****** GX
NMCRS   000124RG            PSIPT   000142RG            SNAPT   000170RG            $CCBAL= ****** GX           $RDBSZ= ****** GX
NMLST   000126RG            P$$P45= 000000              SRSTD   000040RG            $CCBCT= ****** GX           $RDBTH= ****** GX
NMVVCL= 000102 G            P$$WRD= 000000              STATUS  000230RG            $CCBGT= ****** GX           $RELOC= ****** GX
NMVVEC  000000RG            Q$$OPT= 000010              STPCT   000042RG            $CCBNM= ****** GX           $SDBAF= ****** GX
NM.CLN  003400              RBUFF = ****** GX           S$$WRG= 000000              $CCBRT= ****** GX           $SDBCT= ****** GX
NM.INI  002400              RDBAF   000144RG            S$$YSZ= 007600              $CEACC= ****** GX           $SDBNM= ****** GX
NM.OPR  003000              RDBCT   000146RG            TKTCB   000044RG            $COPT = ****** GX           $SLTMA= ****** GX
NM.VR2  002401              RDBNM   000150RG            TSKLN2  000222RG            $CRUNM  000206RG            $SLTNM= ****** GX
NM.VR3  002402              RDBSZ   000152RG            T$$KMG= 000000              $CSBRT= ****** GX           $SNAPT= ****** GX
NTLPT   000130RG            RDBTH   000154RG            T$$MIN= 000000              $DDMAN= ****** GX           $SRSTD= ****** GX
NUMUSR  000220RG            RELOC   000036RG            T.NAM = ****** GX           $DEACB= ****** GX           $STPCT= ****** GX
N$$ACC= 000001              RF$END= 000200 G            T.RCVL= ****** GX           $DECPT= ****** GX           $TKTCB= ****** GX
N$$BUF= 000001              RF$NCX= 000002 G            T.ST3 = ****** GX           $DREXT= ****** GX           $XAVL = ****** GX
N$$LDV= 000001              RF$PRV= 000001 G            T.UCB = ****** GX           $EXRQN= ****** GX           $XBIAS= ****** GX
N$$MCP= 000001              RID   = ****** GX           T3.PRV= ****** GX           $EXRQP= ****** GX           $ZTIME= ****** GX
N$$MLL= 000001              R$FLAG  000002 G            UA.ACC= 000001              $FILHD= ****** GX           $ZTIM2= ****** GX
N$$MOV= 000010              R$FUNC  000000 G            UA.ALL= 000400              $FMASK= ****** GX           $$MSG = 000000
N$$NCT= 000001              P$INIT  000004 G            UA.CAL= 000100              $HEADR= ****** GX           $$$SYS= 004374
N$$PEM= 000001              R$OPER  000006 G            UA.COM= 000200              $IOFIN= ****** GX           $$$SYX= 000000
OBJHD   000132RG            R$PC  = 000016 G            UA.ECH= 000004              $LDBAF= ****** GX           .$$$$.= 000034
O.AST   000006 G            R$R0  = 000002 G            UA.PRO= 000002              $LDBGT= ****** GX           ...GBL= 000000
O.EFN   000010 G            R$R1  = 000004 G

. ABS.  177776     000   (RW,I,GBL,ABS,OVR)
        002760     001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 223
Work  file writes: 186
Size of work file: 38064 Words  ( 149 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
```

```
                                                                                      M 4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-6
Symbol table
```

```
                                                .SBTTL  DLXAST - DLX QIO COMPLETION AST HANDLING
                                        ;+
                                        ; **-DLXAST-DLX QIO COMPLETION AST HANDLING
                                        ;
                                        ; THIS ROUTINE IS CALLED WHEN A DLX QIO HAS BEEN COMPLETED.
                                        ; IT MARKS THE THREAD AS BEING READY TO BE RESUMED.
                                        ;
                                        ; INPUTS:
                                        ;       (SP) = ADDRESS OF I/O STATUS BLOCK IN THREAD CONTEXT AREA (C$STAT)
                                        ;
                                        ; OUTPUTS:
                                        ;       STACK CLEANED OFF, AND
                                        ;       CF$FDN IS SET IN C$FLAG OF CONTEXT AREA TO INDICATE FORK IS DONE
                                        ;       FDNCNT IS INCREMENTED TO INDICATE A THREAD IS READY TO BE RESUMED.
                                        ;
                                        ; REGISTERS:
                                        ;       NO REGISTERS ARE MODIFIED
                                        ;-

000054                                  DLXAST::
000054  010046                                  MOV     R0,-(SP)                ; SAVE R0
000056  016600  000002                          MOV     2(SP),R0                ; GET I/O STATUS BLOCK ADDRESS
000062  052760  000000G 000000C                 BIS     #CF$FDN,C$FLAG-C$STAT(R0); MARK THREAD AS RESUMEABLE
000070  011600                                  MOV     (SP),R0                 ; RESTORE R0
000072  000754                                  BR      ASTX1                   ; AND EXIT THE AST
```

```
                                        .SBTTL  DEACTX - DEALLOCATE CONTEXT AREA
                                ;+
                                ; **-DEACTX-DEALLOCATE A CONTEXT AREA
                                ;
                                ; THIS ROUTINE DEALLOCATES THE CURRENT CONTEXT AREA, AND THE CURRENT
                                ; WINDOW BLOCK IF NEEDED.
                                ;
                                ; INPUTS:
                                ;       CURCTX = ADDRESS OF CONTEXT AREA TO BE DEALLOCATED
                                ;       CURIOP = ADDRESS OF I/O PACKET
                                ;       TSKLN2 = 2ND LUN WORD FROM USER TASK HEADER
                                ;
                                ; OUTPUTS:
                                ;       NUMUSR IS DECREMENTED TO INDICATE ONE LESS USER
                                ;
                                ; REGISTERS:
                                ;       NO REGISTERS ARE MODIFIED
                                ;-

000264                          DEACTX::
000264                                  CALL    $SAVAL                  ;; SAVE ALL REGISTERS
000270  016705  000000G                 MOV     CURCTX,R5               ;; POINT TO CURRENT CONTEXT AREA
000274  001422                          BEQ     10$                     ;; IF EQ, THERE IS NONE

000276  005367  000000G                 DEC     NUMUSR                  ;; ... AND ONE LESS USER
000302  100420                          BMI     20$                     ;; IF MI, TROUBLE       *** DEBUG ***

000304  005065  000000G                 CLR     C$FLAG(R5)              ;; ELSE, CONTEXT IS NOT IN USE
000310  005067  000000G                 CLR     CURCTX                  ;; ... NO CURRENT CONTEXT

000314  016703  000000G                 MOV     CURIOP,R3               ;; GET I/O PACKET ADDRESS
000320  016700  000000G                 MOV     TSKLN2,R0               ;; GET WINDOW BLOCK ADDRESS
000324  001406                          BEQ     10$                     ;; IF EQ, NO BLOCK
000326  005067  000000G                 CLR     TSKLN2                  ;; CLEAR 2ND LUN WORD
000332  012701  000000G                 MOV     #W$SIZ,R1               ;; SET SIZE OF BLOCK
000336                                  CALL    @DDEACB                 ;; DEALLOCATE THE BLOCK         ; RJK02
000342                          10$:    RETURN

000344  000004                  20$:    IOT                             ;; OOPS                 *** DEBUG ***
```

```
                000020                          MXID  = 16.             ; Maximum network management id string length
                000020                          MXDSTN = MXID           ; Max destination name
                000020                          MXGRPN = MXID           ; Max group name
                000020                          MXOBJN = MXID           ; Max object name

                                        ;
                                        ; MODULE entity operation status
                                        ;
        000000                          M$OFLG: .BLKW   1               ; Operation flags
        000002                          M$FLAG: .BLKW   1               ; Flags word 1
        000004                          M$FLG2: .BLKW   1               ; Flags word 2
        000006                          M$ENT:  .BLKW   1               ; Module to operate on
        000010                          M$PAR:  .BLKW   1               ; Parameter value
        000012                          M$OPT:  .BLKW   1               ; Saved options word
        000014                          M$BUF:  .BLKW   1               ; Saved address of input buffer

                                        ;
                                        ; MODULE entity parameter values (common to READ and CHANGE)
                                        ;
        000016                          M$SCO:  .BLKW   1               ; Scope (0 = global)
        000020                          M$DTEL: .BLKW   1               ; Length of DTE address (number of digits)
        000022                          M$DTE:  .BLKB   8.              ; DTE address (BCD)
        000032                          M$DSTL: .BLKB   1               ; Length of destination name
        000033                          M$DST:  .BLKB   MXID            ; Destination name
        000053                          M$GROL: .BLKB   1               ; Length of group name
        000054                          M$GRO:  .BLKB   MXID            ; Group name
                                                .EVEN

                                        ;
                                        ; MODULE entity parameter values (CHANGE only)
                                        ;
        000074                          M$STA:  .BLKW   1               ; New module state
        000076                          M$PCT:  .BLKW   1               ; Counter timer
        000100                          M$DTEP: .BLKW   1               ; Pointer to DTE descriptor
        000102                          M$PMC:  .BLKW   1               ; DTE maximum circuits
        000104                          M$GNM:  .BLKW   1               ; Group number
        000106                          M$GTY:  .BLKW   1               ; Group type
        000110                          M$OBNU: .BLKW   1               ; Object number
        000112                          M$OBNR: .BLKW   2               ; Object name (RAD50)
        000116                          M$PRI:  .BLKW   1               ; Priority
        000120                          M$CALL: .BLKW   1               ; Length of call mask/call value (bytes)
        000122                          M$CMK:  .BLKB   16.             ; Call mask (hex image)
        000142                          M$CVA:  .BLKB   16.             ; Call value (hex image)
        000162                          M$SAD:  .BLKW   2               ; Subaddress
        000166                          M$TXT:  .BLKW   1               ; Address of error text
        000170                          M$PCHC: .BLKW   1               ; Number of channel ranges
        000172                          M$PCH:  .BLKW   12.             ; Room for six channel ranges
        000222                          M$PCHE:                         ; End of channel range table
        000222                          M$BDF:  .BLKW   1               ; Default data
        000224                          M$BMX:  .BLKW   1               ; Max data
        000226                          M$WDF:  .BLKB   1               ; Default window
        000227                          M$WMX:  .BLKB   1               ; Max window
        000230                          M$CAT:  .BLKB   1               ; Call timer
        000231                          M$CLT:  .BLKB   1               ; Clear timer
        000232                          M$RST:  .BLKB   1               ; Reset timer
        000233                          M$RRT:  .BLKB   1               ; Restart timer
        000234                          M$LMX:  .BLKB   1               ; Max clears
        000235                          M$RMX:  .BLKB   1               ; Max resets
        000236                          M$SMX:  .BLKB   1               ; Max restarts
        000237                          M$NETL: .BLKB   1               ; Length of network name
```

```
                                      .SBTTL  MACRO CALLS, LOCAL DEFINITIONS AND STORAGE
                              ;
                              ; MACRO LIBRARY CALLS
                              ;
                                      .MCALL  TCBDF$,PKTDF$

000000                                TCBDF$                          ; DEFINE TCB OFFSETS
000000                                PKTDF$                          ; DEFINE I/O PACKET OFFSETS
                              ;
                              ; OVERLAY SEGMENT DESCRIPTOR OFFSETS
                              ;
000000                                .ASECT
          000000                      .=0
000000                        S$DISK:                                 ; RELATIVE DISK ADDRESS (BITS 0-11)
000000                        S$STAT: .BLKW   1                       ; SEGMENT STATUS (BITS 12-15)
          040000                SS$NDA=40000                          ; ... SEGMENT HAS NO DISK ALLOCATION
          020000                SS$LOA=20000                          ; ... SEGMENT IS LOADED FROM DISK
          010000                SS$OUT=10000                          ; ... SEGMENT IS NOT LOADED OR NOT MAPPED
000002                        S$LOAD: .BLKW   1                       ; LOAD ADDRESS
000004                        S$SIZE: .BLKW   1                       ; LENGTH IN BYTES
000006                        S$UP:   .BLKW   1                       ; LINK UP
000010                        S$DOWN: .BLKW   1                       ; LINK DOWN
000012                        S$NEXT: .BLKW   1                       ; LINK ACROSS
000014                        S$NAME: .BLKW   2                       ; SEGMENT NAME
000020                        S$WIND: .BLKW   1                       ; WINDOW DESCRIPTOR ADDRESS
000000                                .PSECT
```

```
    108 000024                                     ERROR IE.NLN, ME.DON                                ;NO FILE ON LUN
    109 000030                                     ERROR IE.DAO, ME.SIZ                                ;BUFFER TOO SMALL
    110 000034                                     ERROR IE.PRI, ME.PRI                                ;PRIVILEGE VIOLATION
    111 000040                                     ERROR IE.NST, ME.OPE, <Management ACP not installed>
    112 000044                                     ERROR IE.ABO, ME.OPE, <Management ACP aborted>
    113 000050                                     ERROR IE.NFW, ME.OPE, <Permanent database access not supported>
    114 000054                                     ERROR IE.NDR, ME.RES
    115 000060                                     ERROR IE.OFL, ME.OPE, <NM: device offline>
    116
    117                                                  .IF DF R$$PRO   ; PRO/DECnet
    118
    119                                            ERROR IE.RSU, ME.OPE, <Parameter updated but not saved in file>
    120
    121                                                  .ENDC   ; DF R$$PRO
    122
    123 000064                                     ERROR 0,      ME.OPE, <Directive failed - return = >
    124
```

```
                                                                          L 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL  VALUE              REFERENCES

ME$PRO    000005             7-309      7-313
ME$SYS    000006             7-393      7-397      7-401
ME.DON    177600             5-108
ME.OPE    177747             5-106      5-107      5-111      5-112      5-113      5-115      5-123
ME.PRI    177775             5-110
ME.RES    177761             5-114
ME.SIZ    177774             5-109
MF$CHA    000023             7-262      7-280      7-295      7-310      7-328      7-340      7-357      7-378      7-394
MF$REA    000024             7-266      7-273      7-284      7-299      7-314      7-332      7-344      7-361      7-382
                             7-398
MF$ZER    000025             7-288      7-303      7-348      7-365      7-402
MOCHIN    000552 RG         #7-338
MOCHOP    000722 RG         #7-425
MOID      000572 R           7-341      7-345     #7-349
MOREIN    000560 RG         #7-342
MOREOP    000722 RG         #7-426
MOZEIN    000566 RG         #7-346
MOZEOP    000722 RG         #7-427
MXMES   = 000454            #5-62       6-158      6-161
NMSTAT    000010 R          #5-93      *6-155      6-158      6-161     *6-163     *6-164      6-165      6-176     *6-178
                             6-186     *6-192      6-196      6-244
NM.INI    002400             6-158
NM.OPR    003000             6-161
NOCHIN    000600 RG         #7-355
NOCHOP    000722 RG         #7-428
NOID      000620 R           7-358      7-362     #7-366
NOREIN    000606 RG         #7-359
NOREOP    000722 RG         #7-429
NOZEIN    000614 RG         #7-363
NOZEOP    000722 RG         #7-430
OBCHIN    000640 RG         #7-376
OBCHOP    000722 RG         #7-431
OBID      000652 R           7-379     #7-384
OBREIN    000646 RG         #7-380
OBREOP    000722 RG         #7-432
OPRQIO    000156 R          #6-161      7-438
PRCHIN    000506 RG         #7-308
PRCHOP    000722 RG         #7-433
PRID      000520 R           7-311     #7-318
PRREIN    000514 RG         #7-312
PRREOP    000722 RG         #7-434
R$$PRO  = ******             5-65       5-95       5-117      6-198      6-237
SYCHIN    000666 RG         #7-392
SYCHOP    000722 RG         #7-435
SYID      000706 R           7-395      7-399     #7-403
SYREIN    000674 RG         #7-396
SYREOP    000722 RG         #7-436
SYZEIN    000702 RG         #7-400
SYZEOP    000722 RG         #7-437
$CBTA   = ******   GX        6-188
$DSW    = ******   GX        6-163
$TMEFN  = ******   GX        6-158      6-161
```

```
                                                                          M 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00
```

```
ML$TOP   000001     MP$BRT   001620     MP$EVE   000311     MP$MRO   001605     MP$RET   001631
MN$UNL   000377     MP$BSA   002176     MP$FNC   001752     MP$MRP   006322     MP$RFA   001323
MO$ACC   000200     MP$BSD   002203     MP$GDT   002222     MP$MRT   002153     MP$RMX   002201
MO$ADD   000002     MP$BSI   002200     MP$GNM   002223     MP$MRV   006320     MP$RPA   004374
MO$ALA   000004     MP$BSP   006331     MP$GRO   002115     MP$MSB   000170     MP$RPR   001606
MO$ALI   000000     MP$BUF   000170     MP$GRP   000541     MP$MVE   000145     MP$RRT   002213
MO$CHA   000040     MP$BUP   006324     MP$GTY   002224     MP$MVI   001634     MP$RST   002212
MO$CIR   000003     MP$BUS   001643     MP$HAD   001757     MP$MVR   001751     MP$RSV   001754
MO$CLE   000100     MP$CAC   001750     MP$HBT   002142     MP$MWN   002154     MP$RSZ   001756
MO$COU   000060     MP$CAS   001762     MP$HDD   000162     MP$MXB   002172     MP$RTI   001616
MO$DAC   000000     MP$CAT   002210     MP$HTM   001612     MP$MXC   000466     MP$RTM   001611
MO$DEF   000000     MP$CCS   004406     MP$HWA   002210     MP$MXR   001630     MP$RVE   001604
MO$DPR   000000     MP$CHN   002141     MP$IAT   001322     MP$MXW   002143     MP$RVT   000156
MO$ENT   000017     MP$CIR   000144     MP$IDE   000144     MP$NAA   006335     MP$SAC   000514
MO$EVE   000100     MP$CLK   002131     MP$IDP   006327     MP$NAC   001130     MP$SAD   000543
MO$INF   000160     MP$CLN   002126     MP$IHO   000215     MP$NAP   006326     MP$SCA   000310
MO$INS   000002     MP$CLT   002211     MP$INA   002177     MP$NCT   000240     MP$SCO   000144
MO$LIN   000001     MP$CMB   002142     MP$IND   002204     MP$NET   002114     MP$SCT   002176
MO$LOG   000001     MP$CMK   000537     MP$INI   002201     MP$NLI   000765     MP$SDU   000203
MO$MIR   000031     MP$CMX   002153     MP$ITI   000776     MP$NNA   000764     MP$SDV   000160
MO$NAM   000001     MP$CND   000310     MP$LAA   000231     MP$NOD   000500     MP$SEH   004401
MO$NIC   000023     MP$CNU   001753     MP$LAN   000234     MP$NRA   006334     MP$SER   000144
MO$NOD   000000     MP$COB   000311     MP$LAR   000202     MP$NRB   000214     MP$SET   000000
MO$OFF   000001     MP$CON   002126     MP$LAS   000012     MP$NSA   001617     MP$SGZ   001644
MO$ON    000000     MP$COS   001604     MP$LBS   000203     MP$NTI   002141     MP$SID   000176
MO$OPT   000001     MP$CPF   000230     MP$LCO   000226     MP$NUM   001642     MP$SIN   000310
MO$PER   000200     MP$CPL   000232     MP$LCT   000156     MP$NVE   001274     MP$SLI   000156
MO$PRO   000002     MP$CPT   000226     MP$LHL   000232     MP$NXN=  001476     MP$SLO   000171
MO$PRS   000001     MP$CPU   000161     MP$LLE   000227     MP$OAC   000632     MP$SMX   002202
MO$REA   000200     MP$CSZ   001755     MP$LLO   004432     MP$OAN   000620     MP$SND   000500
MO$SET   000000     MP$CUS   002127     MP$LMB   002152     MP$OCO   000776     MP$SNP   006332
MO$STA   000020     MP$CVA   000540     MP$LMX   002200     MP$OHO   000214     MP$SNU   000542
MO$SUM   000000     MP$DAL   002570     MP$LNA   000144     MP$ONA   000764     MP$SOB   000524
MO$TSK   000000     MP$DCO=  001464     MP$LNO   000233     MP$ONR   002114     MP$SPA   000157
MO$VOL   000000     MP$DDT   002177     MP$LOA   000170     MP$OQL   001453     MP$SPR   000536
MO$ZER   000000     MP$DEL   001131     MP$LOG   004514     MP$OTI   000777     MP$SPS   000513
MP$ACB   000012     MP$DES   000156     MP$LOO   000620     MP$OTY   001022     MP$STA   000000
MP$ACC   000514     MP$DEV   002114     MP$LTM   001613     MP$OUS   000777     MP$STI   002140
MP$ACT   004526     MP$DFA   001320     MP$LTY   004533     MP$OVE   001010     MP$STT   002201
MP$ADD   000766     MP$DHO=  001465     MP$LWI   000230     MP$OWN   004374     MP$STY   000175
MP$ADJ   001440     MP$DIA   000173     MP$MAC   004420     MP$PAR   000036     MP$SUB   000001
MP$ADP   006325     MP$DLB   002571     MP$MAD   001630     MP$PAS   000513     MP$SUR   000156
MP$ADS   004406     MP$DLG   004521     MP$MAP   006323     MP$PCH   002152     MP$SUS   000512
MP$ALB   000036     MP$DLI=  001466     MP$MAR   001635     MP$PCO   000024     MP$SVR   000163
MP$AMC   001640     MP$DLT   002200     MP$MAV   006321     MP$PCT   000144     MP$SWI   002260
MP$AMH   001641     MP$DRO   001441     MP$MBN   001636     MP$PDT   002114     MP$SYP   002424
MP$ANB   004420     MP$DST   000454     MP$MBR   001637     MP$PHA   000170     MP$TFL   000156
MP$ASB   000024     MP$DTE   002140     MP$MBU   001642     MP$PHY   000012     MP$TH1   002206
MP$ASC   004432     MP$DTY=  001452     MP$MCB   000156     MP$PLI   000025     MP$TH2   002205
MP$AUS   000512     MP$DUA   000207     MP$MCO   001632     MP$PLN   002140     MP$TH3   002202
MP$BBT   002165     MP$DUC   000210     MP$MDE   004421     MP$PLO   000012     MP$TLN   000202
MP$BDF   002164     MP$DUM   000202     MP$MHO   001633     MP$PMC   004374     MP$TLO   000172
MP$BFQ   002121     MP$DUP   002127     MP$MLB   000202     MP$PNT   002126     MP$TPA   004375
MP$BLK   001616     MP$DVC   002114     MP$MLK   001306     MP$PRI   004411     MP$TRI   002164
MP$BLO   001452     MP$DWE   001321     MP$MLN   001631     MP$PRO   002130     MP$TST   000144
MP$BMX   002176     MP$ELT   000157     MP$MLP   006333     MP$PSS   001763     MP$TYP   002130
MP$BNP   006330     MP$ETY   001605     MP$MRB   002171     MP$PST   001762     MP$UCS   004407
```

```
                                                                      L 12
MANVCD     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32       PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE               REFERENCES

CM$RSX   = 000200            #5-58         7-137
FE.CEX   = 020000            #5-59         7-120
FMASK    = ******   GX        7-120
INIACP     000000 RG         #6-87
INIBUF   = ******   GX        6-96
INIREQ     000014 RG         #7-119
I$$AS    = ******             5-56
ME$SYS     000006             7-137        7-141
ME.CST     177765             7-139
MF$CHA     000023             7-137
MO$SET     000000             7-137
N$$VCT   = ******             7-131
PDVID    = ******   GX        7-125
PDVTA    = ******   GX        7-127
POOLPT   = ******   GX        6-94
R$$MPL   = ******             5-61         6-88
R$$11D   = ******             5-56
R$$11M   = 000000             5-56
R$$11S   = ******             5-56
X$$MCB   = ******             5-56         5-56
ZF.COU   = 001000            #5-56
ZF.DDM   = 000001            #5-56
ZF.DIA   = 004000            #5-56
ZF.DLC   = 000002            #5-56
ZF.DVP   = 100000            #5-56
ZF.INI   = 040000            #5-56
ZF.KMX   = 000020            #5-56
ZF.LLC   = 000004            #5-56
ZF.LMC   = 000100            #5-56
ZF.MAN   = 020000            #5-56
ZF.MFL   = 000010            #5-56
ZF.MTM   = 000400            #5-56
ZF.MUX   = 000040            #5-56
ZF.PSE   = 002000            #5-56
ZF.SLI   = 010000            #5-56
ZF.TIM   = 000200            #5-56
ZF.X3P   = 000000            #5-56
ZS.ASN   = 100000            #5-56
ZS.BSY   = 140000            #5-56
Z.AVL      000014            #5-56
Z.DAT      000016            #5-56
Z.DSP      000000            #5-56         5-56
Z.FLG      000010            #5-56
Z.LEN    = 000016            #5-56
Z.LLN      000006            #5-56
Z.MAP      000020            #5-56
Z.NAM      000004            #5-56
Z.PCB      000012            #5-56         7-129
Z.SCH      000007            #5-56
$HEADR   = ******   GX        7-131
$MBLUN   = ******   GX        7-146
```

```
                                                                      M 12
MANVCD     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32       PAGE 2
MACRO CROSS REFERENCE                                     CREF    04.00
```

```
                                                                  L 13
MANVTB     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE              REFERENCES

MO$NAM     000001             6-180
MO$SET     000000             6-101     6-108     6-115     6-122     6-131     6-141     6-149     6-156     6-164
MO$ZER     000000             6-110     6-117     6-133     6-143     6-170
NOCHIN   = ******    GX       6-141     6-142
NOCHOP   = ******    GX       6-141     6-142
NOREIN   = ******    GX       6-144
NOREOP   = ******    GX       6-144
NOZEIN   = ******    GX       6-143
NOZEOP   = ******    GX       6-143
OBCHIN   = ******    GX       6-149     6-150
OBCHOP   = ******    GX       6-149     6-150
OBREIN   = ******    GX       6-151
OBREOP   = ******    GX       6-151
PRCHIN   = ******    GX       6-156     6-157
PRCHOP   = ******    GX       6-156     6-157
PRREIN   = ******    GX       6-159
PRREOP   = ******    GX       6-159
RF$END   = ******    GX       6-190
RF$NCX   = ******    GX       6-179     6-180     6-181     6-182     6-185
RF$PRV   = ******    GX       6-101     6-102     6-108     6-109     6-110     6-115     6-116     6-117     6-122
                              6-123     6-131     6-132     6-133     6-141     6-142     6-143     6-149     6-150
                              6-156     6-157     6-164     6-165     6-170     6-182
R$$PRO   = ******             6-126     6-168
R$$RTR   = ******             6-155     6-163     6-184
SYCHIN   = ******    GX       6-164     6-165
SYCHOP   = ******    GX       6-164     6-165
SYREIN   = ******    GX       6-171
SYREOP   = ******    GX       6-171
SYZEIN   = ******    GX       6-170
SYZEOP   = ******    GX       6-170
S$$BAS   = ******             6-97      6-145     6-184
TRAACC   = ******    GX       6-182
TRAADD   = ******    GX       6-181
TRAALI   = ******    GX       6-179
TRANAM   = ******    GX       6-180
```

```
                                                                  M 13
MANVTB     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00
```

```
        .TITLE  ALCHIN - ALIAS CHANGE PARAMETERS INITIALIZE
        .IDENT  /V05.00/
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - ALIAS CHANGE PARAMETERS INITIALIZE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

Table of contents

```
     5-    53    MACRO CALLS AND LOCAL DATA
     6-    73    ALCHOP - ALIAS CHANGE PARAMETERS OPERATE
     7-   119    SETALI - SET AN ALIAS
     8-   267    CLRALI - CLEAR AN ALIAS
     9-   352    DEAANB - DEALLOCATE ALIAS NAME BLOCK
    10-   404    FINDA - FIND A MATCHING ALIAS BLOCK
```

```
MP$NVE   001274                  MP$SEH   004401                  MS$ALO   000007                  MT$MUL   000100                  M$$FCS=  000000
MP$NXN=  001476                  MP$SER   000144                  MS$ASE   000006                  MT$NLE   000017                  M$$MGE=  000000
MP$OAC   000632                  MP$SET   000000                  MS$ATR   000011                  MT$NON   000001                  M$$NET=  000000
MP$OAN   000620                  MP$SGZ   001644                  MS$AUT   000000                  MT$NR4   000005                  M$$OVR=  000000
MP$OCO   000776                  MP$SID   000176                  MS$CLE   000003                  MT$NTY   000060                  NM.CLN   003400
MP$OHO   000214                  MP$SIN   000310                  MS$DED   000004                  MT$OCT   000060                  NM.INI   002400
MP$ONA   000764                  MP$SLI   000156                  MS$DIE   000003                  MT$PHA   000002                  NM.OPR   003000
MP$ONR   002114                  MP$SLO   000171                  MS$DUM   000004                  MT$POI   000000                  NM.VR2   002401
MP$OQL   001453                  MP$SMX   002202                  MS$FAI   000013                  MT$QP2   000010                  NM.VR3   002402
MP$OTI   000777                  MP$SND   000500                  MS$GLO   000000                  MT$ROU   000000                  N$$ACC=  000001
MP$OTY   001022                  MP$SNP   006332                  MS$HOL   000002                  MT$R04   000004                  N$$BUF=  000001
MP$OUS   000777                  MP$SNU   000542                  MS$INA   000002                  MT$SEC   000000                  N$$LDV=  000001
MP$OVE   001010                  MP$SOB   000524                  MS$LOA   000003                  MT$SGD   000020                  N$$MCP=  000001
MP$OWN   004374                  MP$SPA   000157                  MS$LOO   000002                  MT$SYS   000002                  N$$MLL=  000001
MP$PAR   000036                  MP$SPR   000536                  MS$OFF   000001                  MT$TER   000001                  N$$MOV=  000010
MP$PAS   000513                  MP$SPS   000513                  MS$ON    000000                  MT$TRI   000002                  N$$NCT=  000001
MP$PCH   002152                  MP$STA   000000                  MS$PRO   000006                  MT$TYP   007777                  N$$PEM=  000001
MP$PCO   000024                  MP$STI   002140                  MS$REA   000004                  MT$USD   000000                  P$$P45=  000000
MP$PCT   000144                  MP$STT   002201                  MS$REF   000001                  MT$X25   000003                  P$$WRD=  000000
MP$PDT   002114                  MP$STY   000175                  MS$RES   000003                  MU$INC   000001                  Q$$OPT=  000010
MP$PHA   000170                  MP$SUB   000001                  MS$RST   000001                  MU$OUT   000002                  R$$DER=  000000
MP$PHY   000012                  MP$SUR   000156                  MS$RSX   000002                  MU$PER   000000                  R$$K11=  000001
MP$PLI   000025                  MP$SUS   000512                  MS$RT    000005                  MV$III   000000                  R$$SND=  000000
MP$PLN   002140                  MP$SVR   000163                  MS$SER   000002                  MV$IV    000001                  R$$11M=  000000
MP$PLO   000012                  MP$SWI   002260                  MS$SHU   000002                  MX$ACT   000020                  SETALI   000072R
MP$PMC   004374                  MP$SYP   002424                  MS$SRV   000007                  MX$CIR   000020                  SF$ANM   000001
MP$PNT   002126                  MP$TFL   000156                  MS$STA   000000                  MX$CLN   000020                  SF$DNM   000002
MP$PRI   004411                  MP$TH1   002206                  MS$SYN   000012                  MX$CNM   000006                  S$$WRG=  000000
MP$PRO   002130                  MP$TH2   002205                  MS$TER   000001                  MX$CON   000006                  S$$YSZ=  007600
MP$PSS   001763                  MP$TH3   002202                  MS$TOP   000003                  MX$DAC   000020                  T$$KMG=  000000
MP$PST   001762                  MP$TLN   000202                  MS$TRI   000005                  MX$DTE   000020                  T$$MIN=  000000
MP$RET   001631                  MP$TLO   000172                  MS$UNR   000005                  MX$FIL   000034                  V$$CTR=  001000
MP$RFA   001323                  MP$TPA   004375                  MS$VMS   000004                  MX$LGC   000777                  W.BATT   000006
MP$RMX   002201                  MP$TRI   002164                  MS.DON   177600                  MX$LIN   000020                  W.BFPD   000014
MP$RPA   004374                  MP$TST   000144                  MS.MOR   000002                  MX$LON   000006                  W.BHVR   000004
MP$RPR   001606                  MP$TYP   002130                  MS.PAR   000003                  MX$NMS   000454                  W.BLGH   000020
MP$RRT   002213                  MP$UCS   004407                  MS.SUC   000001                  MX$NOD   000006                  W.BLPD   000016
MP$RST   002212                  MP$USR   001750                  MT$AR4   000003                  MX$OBJ   000006                  W.BLVR   000002
MP$RSV   001754                  MP$VEC   004410                  MT$ASC   000100                  MX$OWN   000040                  W.BNPD   000015
MP$RSZ   001756                  MP$VER   004406                  MT$BIL   000001                  MX$PAR   000036                  W.BOFF   000012
MP$RTI   001616                  MP$WDF   002165                  MT$BYS   000011                  MX$PAS   000010                  W.BPCB   000000
MP$RTM   001611                  MP$WMX   002177                  MT$CI    000007                  MX$RAC   000047                  W.BSIZ   000010
MP$RVE   001604                  MP$XMT   002166                  MT$CLE   000077                  MX$RID   000047                  X$$DBT=  000000
MP$RVT   000156                  MP$XPF   000242                  MT$COD   000200                  MX$RPS   000047                  $CALLX=  ****** GX
MP$SAC   000514                  MP$XPL   000244                  MT$CON   000001                  MX$SID   000040                  $CRUNM=  ****** GX
MP$SAD   000543                  MP$XPT   000240                  MT$DMC   000004                  MX$SNK   000377                  $PRUNM=  ****** GX
MP$SCA   000310                  MP$XXX   177777                  MT$ETH   000006                  MX$TYP   000077                  $SAVRG=  ****** GX
MP$SCO   000144                  MP$5ST   004406                  MT$HEX   000040                  MX$UID   000020                  $XLINK=  ****** GX
MP$SCT   002176                  MS$ACT   000001                  MT$LPB   000005                  M$$CRB=  000124                  $$$SYS=  004374
MP$SDU   000203                  MS$ADU   000010                  MT$MAX   000037                  M$$CRX=  000000                  $$$SYX=  000000
MP$SDV   000160

. ABS.   000104     000    (RW,I,GBL,ABS,OVR)
         000736     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics
```

```
                                              .SBTTL  SRNAM   - Search For Named Partition
                                      ;+
                                      ;                                      *** SRNAM ***
                                      ;
                                      ; Function:     To search the system PCB list for a named partition.  This
                                      ;               routine is supplied for systems without PLAS support.
                                      ;
                                      ; Call Format:  CALL    SRNAM
                                      ;
                                      ; Inputs:       R3 = address of RAD50 doubleword partition name
                                      ;
                                      ; Outputs:      C-bit set:      partition not found
                                      ;               C-bit clear:    partition found, R2 = PCB address
                                      ;
                                      ; Registers:    R2 destroyed
                                      ;
                                      ;-

001246  017702  000576'               SRNAM:  MOV     @PARHD,R2               ; Get address of first PCB
001252  026213  000004                10$:    CMP     P.NAM(R2),(R3)          ; Match on first word of name?
001256  001004                                BNE     20$                     ; Branch if not
001260  026263  000006  000002                CMP     P.NAM+2(R2),2(R3)       ; Match on second word of name?
001266  001417                                BEQ     50$                     ; Branch if so

001270  032762  000040  000030        20$:    BIT     #PS.SYS,P.STAT(R2)      ; System-controlled partition?
001276  001410                                BEQ     40$                     ; Branch if not
001300  005762  000010                        TST     P.SUB(R2)               ; Is there another partition?
001304  001403                                BEQ     30$                     ; Branch if not
001306  016202  000010                        MOV     P.SUB(R2),R2            ; Get next subpartition PCB address
001312  000757                                BR      10$                     ; Look for match

001314  016202  000012                30$:    MOV     P.MAIN(R2),R2           ; Get address of main partition PCB
001320  011202                        40$:    MOV     (R2),R2                 ; Get address of next PCB
001322  001353                                BNE     10$                     ; Branch if one there
001324  000261                                SEC                             ; Set C-bit for failure

001326                                50$:    RETURN                          ; Exit

                                              .ENDC                           ; End PLAS conditional
```

```
                                                                M  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 8
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE               REFERENCES

$PDSPL  = ******  GX         8-241
$PDVTA  = ******  GX         8-242
$QRMVF  = ******  GX         8-236
$RDBSZ  = ******  GX         8-243
$SLTMA  = ******  GX         8-244
$STDDM  = ******  GX         8-245
$TKTCB  = ******  GX         8-237
$$$ARG  = 000002            #11-617    11-617   #11-617    11-617   #11-617    11-617   #11-617    11-617   #11-617
                             11-617    11-617   #11-617    11-617    11-617    11-617    11-617    11-617    11-617
                             11-617    11-617    11-617    11-617   #11-617    11-617    11-617    11-617    11-617
                             11-617    11-617    11-617    11-617    11-617    11-617   #11-617    11-617    11-617
                             11-617    11-617    11-617    11-617    11-617    11-617    11-617    11-617
$$$GLB  = ******             8-202     8-203
```

```
                                                                N  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31      PAGE 9
MACRO CROSS REFERENCE                                    CREF   04.00
```

```
001742  126327  000013  000005  40$:    CMPB    I.FCN+1(R3),#NM.INI/256.;; INITIALIZE FUNCTION ?
001750  001017                          BNE     45$                     ;; IF NE, NO
001752  026327  000040  000454          CMP     I.PRM+14(R3),#BUFSIZ    ;; WILL DATA FIT INTO THE LOCAL BUFFER ?
001760  101006                          BHI     43$                     ;; IF HI, NO - ERROR
001762                                  CALL    RBUFF                   ;; COPY THE USER'S BUFFER
001766  026327  000032  000062          CMP     I.PRM+6(R3),#MAXID      ;; WILL ID FIT INTO THE LOCAL ID BUFFER ?
001774  101403                          BLOS    44$                     ;; IF LOS, YES
001776                          43$:    CALL    DAOERR                  ;; ELSE, RETURN DATA OVERRUN ERROR
002002  000446                          BR      90$                     ;; ...
002004                          44$:    CALL    RID                     ;; COPY THE USER'S ID BUFFER
002010                          45$:    RETURN                          ;; RETURN TO USER STATE

                                ;
                                ; BACK FROM SYSTEM STATE, IF NO I/O PACKET, LOOK FOR THE FIRST COMPLETED FORK
                                ;
002012  012700  002206'         50$:    MOV     #100$,R0                ; POINT TO FUNCTION DISPATCH TABLE
002016  016703  176166                  MOV     CURIOP,R3               ; GET I/O PACKET ADDRESS
002022  001003                          BNE     60$                     ; IF NE, THERE IS A PACKET TO PROCESS
002024                                  CALL    FRKDON                  ; ELSE, PROCESS ANY COMPLETED FORKS
002030  000624                          BR      GETPKT                  ; AND LOOK FOR MORE WORK
                                ;
                                ; I/O PACKET FOUND, DISPATCH BASED ON FUNCTION CODE
                                ;
002032  121063  000013          60$:    CMPB    (R0),I.FCN+1(R3)        ; MATCH ON FUNCTION ?
002036  001405                          BEQ     70$                     ; IF EQ, YES - DISPATCH
002040  005710                          TST     (R0)                    ; ELSE, AT END OF TABLE ?
002042  001403                          BEQ     70$                     ; IF EQ, YES - ILLEGAL FUNCTION
002044  062700  000004                  ADD     #4,R0                   ; ELSE, POINT TO NEXT TABLE ENTRY
002050  000770                          BR      60$                     ; ... AND TRY AGAIN
002052                          70$:    CALL    @2(R0)                  ; DISPATCH TO FUNCTION ROUTINE
                                ;
                                ; ENTER SYSTEM STATE TO COPY DATA TO USER BUFFER AND COMPLETE REQUEST
                                ;
002056                                  SWSTK$  50$                     ;; ENTER SYSTEM STATE
002062  005701                          TST     R1                      ;; ELSE, ANYTHING TO RETURN ?
002064  001410                          BEQ     80$                     ;; IF EQ, NO
002066  020163  000030                  CMP     R1,I.PRM+4(R3)          ;; IS THE USER'S BUFFER LARGE ENOUGH ?
002072  101403                          BLOS    75$                     ;; IF LOS, YES
002074                                  CALL    DAOERR                  ;; ELSE, OVERRUN ERROR
002100  000402                          BR      80$                     ;; ...
002102                          75$:    CALL    WBUFF                   ;; ELSE, WRITE THE BUFFER TO USER TASK
002106  005767  176104          80$:    TST     DELFLG                  ;; SHOULD CONTEXT BE DELETED ?
002112  001402                          BEQ     90$                     ;; IF EQ, NO
002114                                  CALL    DEACTX                  ;; DEALLOCATE THE CONTEXT AREA ON ERROR
002120  016767  176066  176066  90$:    MOV     CURCTX,LSTCTX           ;; MAKE CURRENT CONTEXT ''LAST'' ACCESSED
002126  005063  000042                  CLR     I.PRM+16(R3)            ;; CLEAR RMS BLOCK LOCK INDICATOR
                                ;
                                ; UNLINK PACKET BEING COMPLETED FROM RECEIVE QUEUE
                                ;
002132  005363  000004                  DEC     I.TCB(R3)               ;; CLEAR ''PACKET BEING PROCESSED'' FLAG
002136  017704  175702                  MOV     @TKTCB,R4               ;; GET OUR TCB ADDRESS          ; RJK02
002142  062704  000000G                 ADD     #T.RCVL,R4              ;; POINT TO THE RECEIVE QUEUE
002146  010405                          MOV     R4,R5                   ;; ...
002150  021403                  95$:    CMP     (R4),R3                 ;; IS THE CURRENT PACKET THE NEXT IN LIST ?
002152  001402                          BEQ     97$                     ;; IF EQ, YES - UNLINK IT
002154  011404                          MOV     (R4),R4                 ;; ELSE, GET ADDRESS OF NEXT
002156  000774                          BR      95$                     ;; AND KEEP LOOKING
002160  011314                  97$:    MOV     (R3),(R4)               ;; UNLINK PACKET FROM LIST
```

```
                                                                                         M  4
MANACP - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:28  Page 11-6
Symbol table

Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:44.39
SY:MANACP.V2 [135,134]MANACP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANACP
```

```
                                                                     N  4
MANACP      CREATED BY  MACRO  ON 29-JUN-85 AT 10:29       PAGE 1

SYMBOL CROSS REFERENCE                                     CREF   04.00
```

```
                                          .SBTTL  MRKAST -MARK TIME COMPLETION AST HANDLING
                                 ;+
                                 ; **-MRKAST-MARK TIME COMPLETION AST HANDLING
                                 ;
                                 ; THIS ROUTINE IS CALLED WHEN A MARK TIME HAS EXPIRED.
                                 ; IT MARKS THE THREAD AS BEING READY TO BE RESUMED.
                                 ;
                                 ; INPUTS:
                                 ;        (SP) = RETURN ADDRESS PAST LINKAGE CODE IN CONTEXT AREA (C$LOC+4)
                                 ;
                                 ; OUTPUTS:
                                 ;        STACK CLEANED OFF, AND
                                 ;        CF$FDN IS SET IN C$FLAG OF CONTEXT AREA TO INDICATE FORK IS DONE
                                 ;        FDNCNT IS INCREMENTED TO INDICATE A THREAD IS READY TO BE RESUMED.
                                 ;
                                 ; REGISTERS:
                                 ;        NO REGISTERS ARE MODIFIED
                                 ;-

000074                           MRKAST::
000074  010046                            MOV     R0,-(SP)                      ; SAVE R0
000076  016600  000002                    MOV     2(SP),R0                      ; GET RETURN ADDRESS
000102  052760  000000G 000000C           BIS     #CF$FDN,C$FLAG-<C$LOC+4>(R0); MARK THREAD AS RESUMEABLE
000110  000744                            BR      ASTX                          ; UNSTOP TASK AND EXIT THE AST

        000001                            .END
```

```
                                        .SBTTL  UPDCTX - UPDATE THE CURRENT CONTEXT
                                ;+
                                ; **-UPDCTX-UPDATE THE CURRENT CONTEXT
                                ;
                                ; THIS ROUTINE IS CALLED TO MAKE SURE THAT THE CONTEXT AT "CONTEX"
                                ; IS THE SAME AS THE ONE POINTED TO BY "CURCTX". IF THE LAST CONTEXT
                                ; ACCESSED (AS INDICATED BY "LSTCTX") IS THE SAME AS THE CONTEXT TO
                                ; BE RESTORED, THEN NOTHING HAS TO BE DONE. ELSE, THE LAST CONTEXT
                                ; HAS TO BE SAVED, AND THE CURRENT CONTEXT MOVED INTO "CONTEX".
                                ;
                                ; INPUTS:
                                ;       CURCTX = ADDRESS OF CURRENT CONTEXT
                                ;       LSTCTX = ADDRESS OF LAST CONTEXT REFERENCED
                                ;
                                ; OUTPUTS:
                                ;       CONTEX = CURRENT CONTEXT, LAST CONTEXT SAVED IF NECCESSARY
                                ;
                                ; REGISTERS:
                                ;       NO REGISTERS ARE MODIFIED
                                ;
                                ;-

 000346                         UPDCTX::
 000346                                 CALL    $SAVAL                  ; SAVE ALL REGISTERS
 000352 012700  000000G                 MOV     #CONTEX,R0              ; POINT TO LOW CORE CONTEXT
 000356 016701  000000G                 MOV     LSTCTX,R1               ; GET LAST CONTEXT ADDRESS
 000362 001407                          BEQ     10$                     ; IF EQ, NO PREVIOUS CONTEXT
 000364 020167  000000G                 CMP     R1,CURCTX               ; IS IT THE SAME AS DESIRED CONTEXT ?
 000370 001420                          BEQ     40$                     ; IF EQ, YES - NO ACTION REQUIRED
 000372 062701  000000G                 ADD     #C$DATA,R1              ; ELSE, POINT PAST LINK AND FLAGS
 000376                                 CALL    20$                     ; SAVE LAST CONTEXT

 000402 016700  000000G         10$:    MOV     CURCTX,R0               ; GET CURRENT CONTEXT ADDRESS
 000406 001411                          BEQ     40$                     ; IF EQ, NO CONTEXT TO LOAD
 000410 062700  000000G                 ADD     #C$DATA,R0              ; POINT PAST LINK AND FLAGS
 000414 012701  000000G                 MOV     #CONTEX,R1              ; POINT TO LOW CORE CONTEXT

 000420 012702  000000C         20$:    MOV     #CTXSIZ/2,R2            ; SET LENGTH OF CONTEXT IN WORDS
 000424 012021                  30$:    MOV     (R0)+,(R1)+             ; SAVE/RESTORE CONTEXT
 000426                                 SOB     R2,30$                  ; LOOP TILL MOVED
 000432                         40$:    RETURN
```

```
      000240                    M$NET:  .BLKB   MXID            ; Network name
      000260                    M$SNDL: .BLKB   1               ; Length of node name
      000261                    M$SND:  .BLKB   6               ; Node name
      000267                    M$OBNL: .BLKB   1               ; Length of object name
      000270                    M$OBNA: .BLKB   MXID            ; Object name (ASCII)
      000310                    M$PLNL: .BLKB   1               ; Length of line name
      000311                    M$PNL:  .BLKB   MXID            ; Line name
      000331                    M$SACL: .BLKB   1               ; Account length
      000332                    M$SAC:  .BLKB   MXACC           ; Account name
      000352                    M$SPSL: .BLKB   1               ; Password length
      000353                    M$SPS:  .BLKB   MXPAS           ; Password
      000363                    M$SUSL: .BLKB   1               ; User name length
      000364                    M$SUS:  .BLKB   MXUID           ; User name
                                        .EVEN
      000404                    M$SCR:  .BLKW   15.             ; Scratch space
      000442                    M$LEN:                          ; Length of context area
                                ;
                                ; Define function bits for operations flag word
                                ;
              000001            MM$DON = 1                      ; Processing complete
              000002            MM$PER = 2                      ; Indicate error detail specified
              000004            MM$KNO = 4                      ; KNOWN specified
              000010            MM$ALL = 10                     ; ALL specified
              000020            MM$OVR = 20                     ; Context area buffer overflowed
              000040            MM$ZRO = 40                     ; ZERO operation in progress
              000100            MM$MUX = 100                    ; Line is a MUX line
                                ;
                                ; Define flags in M$FLAG (parameter presence indicators)
                                ;
                                ;
                                ; Common parameters
                                ;
              000001            MF$DST = 1                      ; DESTINATION specified
              000002            MF$NET = 2                      ; NETWORK specified
                                ;
                                ; Module X25-ACCESS
                                ;
              000010            MF$ANB = 10                     ; NUMBER specified
              000020            MF$ASC = 20                     ; SCOPE specified
                                ;
                                ; Module X25-PROTOCOL (general)
                                ;
              000010            MF$BDF = 10                     ; DEFAULT DATA specified
              000020            MF$BMX = 20                     ; MAX DATA specified
              000040            MF$WDF = 40                     ; DEFAULT WINDOW specified
              000100            MF$WMX = 100                    ; MAX WINDOW specified
              000200            MF$CAT = 200                    ; CALL TIMER specified
              000400            MF$CLT = 400                    ; CLEAR TIMER specified
              001000            MF$RST = 1000                   ; RESET TIMER specified
              002000            MF$RRT = 2000                   ; RESTART TIMER specified
              004000            MF$LMX = 4000                   ; MAX CLEARS specified
              010000            MF$RMX = 10000                  ; MAX RESETS specified
              020000            MF$SMX = 20000                  ; MAX RESTARTS specified
                                ;
                                ; Module X2n-SERVER
                                ;
              000010            MF$SOB = 10                     ; OBJECT specified
```

```
     77                                                   .SBTTL  FORK - SAVE THREAD CONTEXT AND RE-START SCAN
     78                                          ;+
     79                                          ; **-FORK-SAVE THREAD CONTEXT AND RE-START SCAN
     80                                          ;
     81                                          ; THIS ROUTINE IS CALLED TO SAVE THE CONTEXT FOR THE CURRENT THREAD,
     82                                          ; INCLUDING THE TASK STACK, IN THE CONTEXT AREA. THE THREAD WILL
     83                                          ; BE RESUMED WHEN THE CF$FDN FLAG IS SET IN C$FLAG OF THE CONTEXT
     84                                          ; BLOCK. THE BUFFER IS WRITTEN TO THE USER'S BUFFER (TO SAVE IT).
     85                                          ;
     86                                          ; INPUTS:
     87                                          ;        CURCTX = ADDRESS OF CURRENT CONTEXT BLOCK
     88                                          ;        CURIOP = ADDRESS OF CURRENT I/O PACKET
     89                                          ;        TSKLN2 = ADDRESS OF TASK'S 2ND LUN WORD
     90                                          ;        ISTACK = INITIAL STACK POINTER FOR TASK
     91                                          ;
     92                                          ; OUTPUTS:
     93                                          ;        CURCTX = ADDRESS OF SAVED CONTEXT AREA WITH
     94                                          ;         C$SP = SAVED STACK POINTER, AND TASK STACK
     95                                          ;         COPIED INTO THE HIGH ADDRESSES IN THE CONTEXT
     96                                          ;         AREA.
     97                                          ;        BUFFER IS MOVED INTO THE USER'S BUFFER
     98                                          ;-
     99
    100 000000                                   FORK::
    101 000000                                           CALL    $SAVAL                  ; SAVE ALL REGISTERS ON CONTEXT STACK
    102                                          ;
    103                                          ; SAVE OVERLAY CONTEXT
    104                                          ;
    105 000004  005005                                   CLR     R5                      ; INIT SAVED SEGMENT DESCRIPTOR COUNT
    106 000006  016704  000000G                          MOV     .NSTBL,R4               ; GET POINTER TO ROOT SEG DESCRIPTOR
    107 000012  010400                           10$:    MOV     R4,R0                   ; POINT TO NEXT ROOT SEGMENT
    108 000014  005205                                   INC     R5                      ; INDICATE ANOTHER SEGMENT SAVED
    109 000016  016001  000006                   20$:    MOV     S$UP(R0),R1             ; GET ADD OF NEXT HIGHER SEGMENT
    110 000022  001413                                   BEQ     50$                     ; IF EQ, NO HIGHER IN THIS TREE
    111 000024  032711  010000                   30$:    BIT     #SS$OUT,(R1)            ; IS THIS MAPPED ?
    112 000030  001406                                   BEQ     40$                     ; IF EQ, YES - LOOK FOR NEXT HIGHER
    113 000032  016101  000012                           MOV     S$NEXT(R1),R1           ; ELSE, SCAN ACROSS TREE
    114 000036  020160  000006                           CMP     R1,S$UP(R0)             ; ARE WE BACK TO WHERE WE STARTED SCAN?
    115 000042  001403                                   BEQ     50$                     ; IF EQ, YES - NO SEGMENT LOADED HERE
    116 000044  000767                                   BR      30$                     ; ELSE, SEE IF NEXT SEG IS LOADED
    117 000046  010100                           40$:    MOV     R1,R0                   ; SAVE THIS AS HIGHEST SEG LOADED SO FAR
    118 000050  000762                                   BR      20$                     ; ... AND KEEP LOOKING ON THIS TREE
    119 000052  010046                           50$:    MOV     R0,-(SP)                ; SAVE HIGHEST LOADED SEG DESCRIPTOR
    120 000054  016404  000012                           MOV     S$NEXT(R4),R4           ; GET ADDRESS OF NEXT CO-TREE ROOT
    121 000060  020467  000000G                          CMP     R4,.NSTBL               ; ARE WE BACK AT THE ROOT ?
    122 000064  001352                                   BNE     10$                     ; IF NE, NO - KEEP SCANNING
    123 000066  010546                                   MOV     R5,-(SP)                ; ELSE, SAVE NUMBER OF SEG DESCRIPTORS
    124                                          ;
    125                                          ; SAVE THE CURRENT I/O PACKET ADDRESS, THE "TI:" AND THE TASK STACK
    126                                          ;
    127 000070  016746  000000G                          MOV     CURIOP,-(SP)            ; SAVE I/O PACKET ADDRESS
    128 000074  016746  000000G                          MOV     TSKLN2,-(SP)            ; SAVE TASK'S 2ND LUN WORD
    129 000100  017701  000000G                          MOV     @TKTCB,R1               ; GET OUR TCB ADDRESS           ; RJK01
    130 000104  016146  000026                           MOV     T.UCB(R1),-(SP)         ; ... AND SAVE OUR TI: UCB ADDRESS
    131 000110  012701  000000C                          MOV     #CONTEX+CTXSIZ,R1       ; POINT TO STACK SAVE AREA IN CONTEXT
    132 000114  012641                           60$:    MOV     (SP)+,-(R1)             ; SAVE TASK STACK ON CONTEXT STACK
    133 000116  020667  000000G                          CMP     SP,ISTACK               ; MORE TO COPY ?
```

```
                                .SBTTL  INIQIO - ISSUE NM.INI QIO TO NM:
                                .SBTTL  OPRQIO - ISSUE NM.OPR QIO TO NM:
                        ;+
                        ; **-INIQIO-ISSUE NM.INI QIO TO NM:
                        ; **-OPRQIO-ISSUE NM.OPR QIO TO NM:
                        ;
                        ; THESE ROUTINES ISSUE THE MANAGEMENT QIOS TO NM:, AND FORMAT
                        ; THE ERROR RETURN CODES.
                        ;
                        ; INPUTS:
                        ;         INIQIO:
                        ;           R0 = LENGTH IF ID STRING
                        ;           R1 = ADDRESS OF ID STRING
                        ;           R2 = LENGTH OF NICE MESSAGE IN BUFFER
                        ;           R3 = ADDRESS OF NICE MESSAGE BUFFER
                        ;           R4 = NICE FUNCTION AND OPTION
                        ;         OPRQIO:
                        ;           R3 = ADDRESS OF NICE MESSAGE BUFFER
                        ;
                        ; OUTPUTS:
                        ;         C-BIT SUCCESS/FAILURE INDICATOR AND:
                        ;           R0 = NICE STATUS CODE
                        ;           R2 = LENGTH OF DATA RETURNED IN BUFFER
                        ;           R3 = BUFFER ADDRESS
                        ;
                        ;-

                                .ENABL  LSB

000070  005067  177716          INIQIO: CLR     NMSTAT+2                        ; NO DATA RETURNED
000074                                  DIR$    #ALUN                           ; ASSIGN LUN TO NM:
000102  103451                          BCS     10$                             ; IF CS, PROCESS ERROR
000104                                  QIOW$S  #NM.INI,#$TMLUN,#$TMEFN,,#NMSTAT,,<R4,R1,R0,R3,R2,#MXMES>
000154  000424                          BR      10$                             ; CHECK AND FORMAT STATUS

000156                          OPRQIO: QIOW$S  #NM.OPR,#$TMLUN,#$TMEFN,,#NMSTAT,,<R3,#MXMES>
000226  103005                  10$:    BCC     20$                             ; IF CC, DIRECTIVE WAS OK
000230  116767  00000'  177552          MOVB    $DSW,NMSTAT                     ; ELSE DSW CONTAINS ERROR
000236  105067  17754,                  CLRB    NMSTAT+1                        ; ZERO TO FALL THRU TO ERROR PROCESSING
000242  105767  177543          20$:    TSTB    NMSTAT+1                        ; SUCCESSFUL RETURN ?
000246  100451                          BMI     70$                             ; IF MI, NO - HAVE MANAGEMENT RESPONSE
000250  001051                          BNE     80$                             ; IF NE, YES
000252  010300                  25$:    MOV     R3,R0                           ; COPY BUFFER POINTER
000254  105020                          CLRB    (R0)+                           ; SKIP OVER RETURN CODE
000256  112720  177777                  MOVB    #MD.NON,(R0)+                   ; NO DETAIL CODE
000262  112720  177777                  MOVB    #MD.NON,(R0)+                   ;
000266  012702  000010'                 MOV     #ERRTBL-4,R2                    ; POINT TO ERROR TABLE
000272  022222                  30$:    CMP     (R2)+,(R2)+                     ; BUMP TABLE POINTER
000274  105712                          TSTB    (R2)                            ; END OF TABLE ?
000276  001403                          BEQ     40$                             ; IF EQ, YES
000300  121267  177504                  CMPB    (R2),NMSTAT                     ; IS THIS THE ERROR CODE ?
000304  001372                          BNE     30$                             ; IF NE, NO
000306  116267  000001  177475  40$:    MOVB    1(R2),NMSTAT+1                  ; MAP IT ONTO NETWORK MGMT RETURN
000314  016201  000002                  MOV     2(R2),R1                        ; CHECK FOR DETAIL STRING
000320  001424                          BEQ     70$                             ; IF EQ, NONE
000322  105020                          CLRB    (R0)+                           ; SKIP OVER LENGTH BYTE
000324  112120                  50$:    MOVB    (R1)+,(R0)+                     ; COPY DETAIL STRING
```

```
                                                                      M 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31        PAGE 3
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL   VALUE              REFERENCES

$TMLUN   = ******   GX       5-92      6-158     6-161
$$$      = 000235 R         #5-106    #5-106     5-106    #5-107    #5-107     5-107    #5-108     5-108    #5-109
                             5-109    #5-110     5-110    #5-111    #5-111     5-111    #5-112    #5-112     5-112
                            #5-113    #5-113     5-113    #5-114     5-114    #5-115    #5-115     5-115    #5-123
                            #5-123     5-123
$$$ARG   = 000002           #6-158     6-158    #6-158     6-158    #6-158     6-158    #6-158     6-158    #6-158
                             6-158    #6-158     6-158    #6-158     6-158    #6-158     6-158     6-158     6-158
                             6-158     6-158     6-158     6-158     6-158     6-158     6-158    #6-158     6-158
                             6-158     6-158     6-158     6-158     6-158     6-158     6-158     6-158     6-158
                            #6-158     6-158     6-158     6-158     6-158     6-158     6-158     6-158     6-158
                             6-158     6-158    #6-161     6-161    #6-161     6-161    #6-161     6-161     6-161
                            #6-161     6-161     6-161     6-161     6-161     6-161     6-161     6-161     6-161
                             6-161     6-161    #6-161     6-161     6-161     6-161     6-161     6-161     6-161
                             6-161     6-161     6-161     6-161    #6-161     6-161     6-161     6-161     6-161
                             6-161     6-161     6-161     6-161     6-161     6-161
$$$GLB   = ******            5-92
$$$T1    = 000002           #5-92
$$$T2    = 000004 R         #5-92      5-92
```

```
                                                                      N 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31        PAGE 4
MACRO CROSS REFERENC                                       CREF    04.00
```

```
                                                                         M 11
MANTRA - TRANSLATE NAMES AND AD MACRO V05.03b Saturday 29-Jun-85 10:31  Page 9-4
Symbol table

MP$USR  001750        MS$REF  000001        MT$MAX  000037        MX$DTE  000020        N$$BUF= 000001
MP$VEC  004410        MS$RES  000003        MT$MUL  000100        MX$FIL  000034        N$$LDV= 000001
MP$VER  004406        MS$RST  000001        MT$NLE  000017        MX$LGC  000777        N$$MCP= 000001
MP$WDF  002165        MS$RSX  000002        MT$NON  000001        MX$LIN  000020        N$$MLL= 000001
MP$WMX  002177        MS$RT   000005        MT$NR4  000005        MX$LON  000006        N$$MOV= 000010
MP$XMT  002166        MS$SER  000002        MT$NTY  000060        MX$NMS  000454        N$$NCT= 000001
MP$XPF  000242        MS$SHU  000002        MT$OCT  000060        MX$NOD  000006        N$$PEM= 000001
MP$XPL  000244        MS$SRV  000007        MT$PHA  000002        MX$OBJ  000006        P$$P45= 000000
MP$XPT  000240        MS$STA  000000        MT$POI  000000        MX$OWN  000040        P$$WRD= 000000
MP$XXX  177777        MS$SYN  000012        MT$QP2  000010        MX$PAR  000036        Q$$OPT= 000010
MP$5ST  004406        MS$TER  000001        MT$ROU  000000        MX$PAS  000010        REPALI= ****** GX
MS$ACT  000001        MS$TOP  000003        MT$RO4  000004        MX$RAC  000047        R$$DER= 000000
MS$ADU  000010        MS$TRI  000005        MT$SEC  000000        MX$RID  000047        R$$K11= 000001
MS$ALO  000007        MS$UNR  000005        MT$SGD  000020        MX$RPS  000047        R$$SND= 000000
MS$ASE  000006        MS$VMS  000004        MT$SYS  000002        MX$SID  000040        R$$11M= 000000
MS$ATR  000011        MS.DON  177600        MT$TER  000001        MX$SNK  000377        SF$ANM  000001
MS$AUT  000000        MS.MOR  000002        MT$TRI  000002        MX$TYP  000077        SF$DNM  000002
MS$CLE  000003        MS.PAR  000003        MT$TYP  007777        MX$UID  000020        S$$WRG= 000000
MS$DED  000004        MS.SUC  000001        MT$USD  000000        M$$CRB= 000124        S$$YSZ= 007600
MS$DIE  000003        MT$AR4  000003        MT$X25  000003        M$$CRX= 000000        TRAACC  000216RG
MS$DUM  000004        MT$ASC  000100        MU$INC  000001        M$$FCS= 000000        TRAADD  000000RG
MS$FAI  000013        MT$BIL  000001        MU$OUT  000002        M$$MGE= 000000        TRAALI  000134RG
MS$GLO  000000        MT$BYS  000011        MU$PER  000000        M$$NET= 000000        TRANAM  000056RG
MS$HOL  000002        MT$CI   000007        MV$III  000000        M$$OVR= 000000        T$$KMG= 000000
MS$INA  000002        MT$CLE  000077        MV$IV   000001        NM.CLN  003400        T$$MIN= 000000
MS$LOA  000003        MT$COD  000200        MX$ACT  000020        NM.INI  002400        V$$CTR= 001000
MS$LOO  000002        MT$CON  000001        MX$CIR  000020        NM.OPR  003000        X$$DBT= 000000
MS$OFF  000001        MT$DMC  000004        MX$CLN  000020        NM.VR2  002401        $HEADR= ****** GX
MS$ON   000000        MT$ETH  000006        MX$CNM  000006        NM.VR3  002402        $$$SYS= 004374
MS$PRO  000006        MT$HEX  000040        MX$CON  000006        N$$ACC= 000001        $$$SYX= 000000
MS$REA  000004        MT$LPB  000005        MX$DAC  000020

. ABS.  000020     000    (RW,I,GBL,ABS,OVR)
        000410     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 197
Work  file writes: 132
Size of work file: 25433 Words  ( 100 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:23.15
SY:MANTRA.V2,[135,134]MANTRA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANTRA
```

```
                                                                    N 11
MANTRA     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31       PAGE 1

SYMBOL CROSS REFERENCE                                    CREF    04.00
```

M 12

```
MANVCD     CREATED BY  MACRO  ON 29-JUN-85 AT 10:32     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME        REFERENCES

ALUN$S            #5-53        7-146
CALL               6-96        7-123        7-125
DIR$              #5-53       #7-146        7-146
GIN$              #5-53
MANDF$            #5-53        5-55
MOV$              #7-146       7-146        7-146        7-146
PDVDF$            #5-53        5-56
RETC              #5-53        7-131
RETURN             6-97        7-132        7-148
SWSTK$             7-123
```

N 12

```
**FILE**ID**MANVTB
```

```
MANVTB      CREATED BY  MACRO  ON 29-JUN-85 AT 10:32     PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

MANDF$            #5-54      5-56
MNFUN$            #5-62      6-95       6-101      6-102      6-103      6-108      6-109      6-110      6-111      6-115
                   6-116     6-117      6-118      6-122      6-123      6-124      6-131      6-132      6-133      6-134
                   6-141     6-142      6-143      6-144      6-149      6-150      6-151      6-156      6-157      6-159
                   6-164     6-165      6-170      6-171      6-179      6-180      6-181      6-182      6-185      6-190
```

```
**FILE**ID**PSIBYE
```

```
                                        .SBTTL  MACRO CALLS AND LOCAL DATA
                                ;
                                ; MACRO LIBRARY CALLS
                                ;
                                        .MCALL  MANDF$,ALCCX$

   000000                               MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS

   000000                               ALCCX$  LIST                    ; DEFINE CONTEXT AREA
                                ;
                                ; CONTEXT AREA OFFSET DEFINITIONS
                                ;
                   000020               MAXID=16.                       ; MAXIMUM LENGTH OF USER ID
                   000010               MAXPAS=8.                       ; MAXIMUM LENGTH OF PASSWORD
                   000020               MAXACC=16.                      ; MAXIMUM LENGTH OF ACCOUNT
         000000                         .ASECT
                   000000               .=0
                                ;
                                ; OFFSETS IN COMMON WITH READ INFORMATION MODULES
                                ;
         000000                 A$FLG:  .BLKW   1                       ; FLAG WORD
         000002                 A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
         000004                 A$PAR:  .BLKW   1                       ; CURRENT PARAMETER TYPE
         000006                 A$BUF:  .BLKW   1                       ; SAVED POINTER TO BUFFER
         000010                 A$NUM:  .BLKB   1                       ; NUMBER OF BYTES IN ANB (FOR SET)
         000011                 A$TYP:  .BLKB   1                       ; ALIAS FORMAT TYPE
         000012                 A$ALI:  .BLKW   3                       ; ALIAS NAME, SPACE FILLED
         000020                 A$SCO:  .BLKW   1                       ; SCOPE OF ALIAS
                                ;
                                ; OFFSETS UNIQUE TO CHANGE PARAMETERS MODULES
                                ;
         000022                 A$DES:  .BLKW   3                       ; DESTINATION NAME, SPACE FILLED
         000030                         .BLKB   <1+MAXID>+<1+MAXPAS>+<1+MAXACC> ; ACCESS CONTROL STORAGE
         000103                 A$LEN:                                  ; LENGTH OF CONTEXT AREA
         000000                         .PSECT
                                ;
                                ; FLAGS WORD BIT DEFINITIONS
                                ;
                   000001               AF.INI=1                        ; SUCCESSFUL INITIALIZATION COMPLETED
                   000002               AF.OVR=2                        ; CONTEXT AREA OVERFLOWED
```

```
	.TITLE	ALCHOP - ALIAS CHANGE PARAMETERS OPERATE
	.IDENT	/V05.00/
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - ALIAS CHANGE PARAMETERS OPERATE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	14-DEC-79
;	DECNET-11M/S V3.0
;	DECNET-11M-PLUS V1.0
;
; 2.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
;
```

```
Work  file  reads: 210
Work  file writes: 145
Size of work file: 28019 Words  ( 110 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:39.38
DB2:ALCHOP,T34,[135,134]ALCHOP/CR/-SP=DB2:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ALCHOP
```

```
                                                .SBTTL  SLOAD   - Load Control for RSX-11S Systems
                                        ;+
                                        ;                               *** SLOAD ***
                                        ;
                                        ; Function:     To control microcode load sequences on RSX-11S systems.
                                        ;               This routine segments a microcode partition into 4K word pieces
                                        ;               for dispatch to the device driver and maintains the load seq-
                                        ;               uence counter(s) and status.
                                        ;
                                        ; Call Format:  CALL    SLOAD
                                        ;
                                        ; Inputs:       PCBADR  = address of partition's PCB
                                        ;               R4 = address of "load request" CCB
                                        ;
                                        ; Outputs:      Load sequence ended
                                        ;               R3 = load halt status
                                        ;               R4 = address of "halt load" CCB
                                        ;
                                        ; Registers:    R0 through R5 are destroyed
                                        ;
                                        ;-

 001330  012700  000001                 SLOAD:  MOV     #NBUFS,R0               ; Set up a loop counter
 001334  012702  000152'                        MOV     #BLOCK,R2               ; Get address of load control words
 001340  062704  000014                         ADD     #C.BUF1,R4              ; Point at first CCB buffer descriptor

 001344  012224                         10$:    MOV     (R2)+,(R4)+             ; Copy bias of next partition segment
 001346  012724  140000                         MOV     #140000,(R4)+           ; Starting address is virtual 0 (APR6)
 001352  012703  000177                         MOV     #177,R3                 ; Assume segment size =177*32 =4k words
 001356  020312                                 CMP     R3,(R2)                 ; Is that greater than what's left?
 001360  003002                                 BGT     20$                     ; Branch if so and do some adjusting
 001362  160312                                 SUB     R3,(R2)                 ; Compute # 32-word blocks left to load
 001364  000402                                 BR      30$                     ; Branch to common code

 001366  011203                         20$:    MOV     (R2),R3                 ; Load segment size = partition remnant
 001370  005012                                 CLR     (R2)                    ; No data left to load

 001372  060342                         30$:    ADD     R3,-(R2)                ; Compute bias of next load segment
         000006                                 .REPT   6                       ; Compute number of bytes to load
                                                ASL     R3                      ;   from number of 32-word blocks
                                                .ENDR                           ;     in current load segment
 001410  010324                                 MOV     R3,(R4)+                ; Move segment size in bytes to CCB
 001412  005024                                 CLR     (R4)+                   ; Clear CCB flag word
 001414  005767  000154'                        TST     BLOCK+2                 ; Has end of partition been reached?
 001420  001402                                 BEQ     40$                     ; Branch if so
 001422                                         SOB     R0,10$                  ; Loop around if more to load

 001426  012744  100000                 40$:    MOV     #CF.LB,-(R4)            ; Mark last descriptor as last buffer
 001432  016704  000156'                        MOV     CURQST,R4               ; Retrieve starting address of CCB
 001436                                         CALL    DSPDRV                  ; Call driver to load microcode
 001442  103404                                 BCS     50$                     ; Branch if driver halted the load
 001444  005767  000154'                        TST     BLOCK+2                 ; Has the entire partition been loaded?
 001450  001327                                 BNE     SLOAD                   ; Branch if not
 001452  005003                                 CLR     R3                        Load halt status = end of data

 001454                                 50$:    RETURN                          ; Exit

```

```
                                                                      N  2
MLD11S     CREATED BY  MACRO  ON 28-AUG-85 AT 11:31     PAGE 9
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME          REFERENCES

ALUN$S              #6-78      9-269
CALL                 9-329     9-330      9-352      9-355      9-361      9-372      9-438      9-455      9-466     10-499
                    10-529    10-574     11-609     12-655     12-661     12-670     12-674     12-680     12-682     16-848
                    16-852    16-864     16-873     16-876     18-980
CALLR               10-547
CCBDF$              #6-79      6-82
DIR$                #6-89     #9-269      9-269     #9-318      9-318     #9-319      9-319      9-340      9-341     #9-469
                     9-469   #11-617     11-617    #12-687     12-687
EVLDF$              #6-79      6-83
EXIT$S              #6-78      9-469     12-687
MOV$                #9-269     9-269      9-269      9-269     #9-318      9-318     #9-319      9-319    #11-617     11-617
                    11-617    11-617     11-617    #11-617     11-617     11-617    #11-617     11-617     11-617
MRKT$               #6-89      8-202
MVB$               #11-617    11-617
OFF$                #8-202    #8-203
PLBDF$              #6-72      6-74
PDVDF$              #6-79      6-84
QDPB$S             #11-617    11-617
QIOW$S              #6-78     11-617
RETURN               9-337     9-369      9-374     10-562     10-577     11-618     12-685     16-887     17-927     18-986
RVP$               #11-617    11-617     11-617     11-617
SLTDF$              #6-79      6-85
SOB                 18-976
SPRA$S              #6-78      9-318
SREA$S              #6-78      9-319
STSE$               #6-89      8-203
TCBDF$              #6-79      6-81
```

```
002162  001002                                  BNE     98$                     ;; IF NE, NOT LAST IN LIST
002164  010465  000002                          MOV     R4,2(R5)                ;; ELSE, CLOSE UP LIST
                                        ;
                                        ; UPDATE TASK'S 2ND LUN WORD, POST COMPLETION ON I/O AND LOOK FOR MORE WORK
                                        ;
002170                                  98$:    .IF DF  R$$MPL
                                                MOV     I.TCB(R3),R5            ;; GET TASK'S TCB ADDRESS
                                                MOV     T.PCB(R5),R5            ;; GET PARTITION CONTROL BLOCK ADDRESS
                                                SAVMAP                          ;; SAVE CURRENT MAPPING
                                                MAP     P.REL(R5)               ;; MAP TO THE TASK HEADER
                                                .IFTF   ;DF R$$MPL
002170  016773  176026  000006                  MOV     TSKLN2,@I.LN2(R3)       ;; UPDATE THE TASK'S SECOND LUN WORD
                                                .IFT    ;DF R$$MPL              ;; ... (CLEARING LUN INTERLOCK BIT)
                                                RESMAP                          ;; RESTORE PREVIOUS MAPPING
                                                .ENDC   ;DF R$$MPL
002176                                          CALL    @IOFIN                  ;; FINISH UP I/O REQUEST        ; RJK02
002202  000167  177300                          JMP     10$                     ;; AND LOOK FOR NEXT
                                        ;
                                        ; I/O FUNCTION CODE DISPATCH TABLE
                                        ;
002206                                  100$:   IOFUN$  NM.INI,MANINI           ; INITIALIZE REQUEST
002212                                          IOFUN$  NM.OPR,MANOPR           ; OPERATE REQUEST
002216                                          IOFUN$  NM.CLN,MANCLN           ; CLEAN UP LUN
002222                                          IOFUN$  0,IFCERR                ; ILLEGAL FUNCTION
```

```
MANACP     CREATED BY  MACRO  ON 29-JUN-85 AT 10:29      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE           REFERENCES

AALOCB     000010 RG      #5-88
ALLCTX   = ******   GX     8-571
ALNERR     002234 R       #7-497       8-539
BIT0     = 000001   G     #5-237
BIT1     = 000002   G     #5-238
BIT10    = 002000   G     #5-247
BIT11    = 004000   G     #5-248
BIT12    = 010000   G     #5-249
BIT13    = 020000   G     #5-250
BIT14    = 040000   G     #5-251
BIT15    = 100000   G     #5-252
BIT2     = 000004   G     #5-239
BIT3     = 000010   G     #5-240
BIT4     = 000020   G     #5-241
BIT5     = 000040   G     #5-242
BIT6     = 000100   G     #5-243
BIT7     = 000200   G     #5-244
BIT8     = 000400   G     #5-245
BIT9     = 001000   G     #5-246
BLXIO      000012 RG      #5-89
BUFFER     000734 RG      #5-276       8-587       9-621
BUFSIZ   = 000454         #5-77        5-275       5-276       6-397
CALLX      000202 RG      #5-169
CCBAF      000046 RG      #5-103
CCBAL      000050 RG      #5-104
CCBCT      000052 RG      #5-105
CCBGT      000054 RG      #5-106
CCBNM      000060 RG      #5-108
CCBRT      000056 RG      #5-107
CEACC      000062 RG      #5-109
CF$BSY   = 000001   G     #5-197
CF$FDN   = 000002   G     #5-198       6-340
CF$VR2   = 000004   G     #5-199       8-575
CF$VR3   = 000010   G     #5-200       8-578
CM$RSX   = 000200         #5-81        8-544
CONTEX     000244 RG      #5-274       8-588       9-622
COPT       000014 RG      #5-90        6-347
CSBRT      000064 RG      #5-110
CTXHD      000232 RG      #5-269
CTXSIZ   = 000466   G     #5-76        5-205       5-273       5-274
CURCTX     000212 RG      #5-261      *6-314       6-438      *9-618      *10-649
CURIOP     000210 RG      #5-260      *6-313      *6-370       6-410       11-693
C$CCB      000012   G     #5-203      *6-339
C$DATA     000032   G     #5-205
C$FLAG     000002   G     #5-196      *6-340      *8-575      *8-578
C$LINK     000000   G     #5-195
C$LOC      000012   G     #5-204
C$SIZE     000520   G     #5-206
C$SP       000004   G     #5-201
C$STAT     000006   G     #5-202
C.ADD      000034          6-330       6-332
C.NSP      000004          6-338
```

```
                                                                                         N 5
MANAST - NETWORK MANAGEMENT ACP MACRO V05.03b Saturday 29-Jun-85 10:29  Page 8-1
Symbol table

ASTX     000022R     D$$BUG= 177514        I$$RDN= 000000        M$$MGE= 000000        Q$$OPT= 000010
ASTX1    000024R     D$$ISK= 000000        K$$CNT= 177546        M$$NET= 000000        REXAST  000046RG
A$$CHK= 000000       D$$L11= 000001        K$$CSR= 177546        M$$OVR= 000000        R$$DER= 000000
A$$CPS= 000000       D$$YNC= 000000        K$$LDC= 000000        NTLAST  000000RG      R$$K11= 000001
A$$PRI= 000000       D$$YNM= 000000        K$$TPS= 000074        N$$ACC= 000001        R$$SND= 000000
A$$TRP= 000000       EXRQN = ****** GX     LD$LP = 000000        N$$BUF= 000001        R$$11M= 000000
CF$FDN= ****** GX    E$$XPR= 000000        L$$ASG= 000000        N$$LDV= 000001        S$$WRG= 000000
C$FLAG= ****** GX    FDNCNT= ****** GX     L$$DRV= 000000        N$$MCP= 000001        S$$YSZ= 007600
C$LOC = ****** GX    F$$LVL= 000001        L$$P11= 000001        N$$MLL= 000001        TKTCB = ****** GX
C$STAT= ****** GX    G$$TPP= 000000        L$$11R= 000000        N$$MOV= 000010        T$$KMG= 000000
C$$CKP= 000000       G$$TSS= 000000        MRKAST  000074RG      N$$NCT= 000001        T$$MIN= 000000
C$$ORE= 000400       G$$TTK= 000000        M$$CRB= 000124        N$$PEM= 000001        V$$CTR= 001000
C$$RSH= 177564       G$$WRD= 000000        M$$CRX= 000000        P$$P45= 000000        X$$DBT= 000000
DLXAST  000054RG     I$$RAR= 000000        M$$FCS= 000000        P$$WRD= 000000

. ABS.  000000    000   (RW,I,GBL,ABS,OVR)
        000112    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 9114 Words  ( 36 Pages)
Size of core pool: 14440 Words  ( 55 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:07.15
SY:MANAST.V2,[135,134]MANAST/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]MANAST
```

```
                                        .SBTTL  RBUFF - READ THE USER'S BUFFER INTO "BUFFER"
                                        .SBTTL  READ - READ A SPECIFIED BUFFER INTO "BUFFER"
                                        .SBTTL  RID - READ THE USER'S ID STRING INTO "IDBUFF"
                                ;+
                                ; **-RBUFF-READ THE USER'S BUFFER INTO "BUFFER"
                                ; **-READ-READ A SPECIFIED BUFFER INTO "BUFFER"
                                ; **-RID-READ THE USER'S ID STRING INTO "IDBUFF"
                                ;
                                ; THIS ROUTINE COPIES THE SUPPLIED BUFFER TO A LOCAL BUFFER
                                ;
                                ; INPUTS:
                                ;       FOR RBUFF AND RID:
                                ;               R3 = ADDRESS OF I/O PACKET WITH BUFFER DESCRIPTOR
                                ;       FOR READ:
                                ;               R3 = ADDRESS OF 3 WORD BUFFER DESCRIPTOR
                                ;
                                ; OUTPUTS:
                                ;       BUFFER IS COPIED
                                ;
                                ; REGISTERS:
                                ;       FOR RBUFF AND RID:
                                ;               NO REGISTERS MODIFIED
                                ;       FOR READ:
                                ;               R0, R1, R2 AND R3 ARE MODIFIED
                                ;-

                                        .ENABL  LSB

000434                          RBUFF:: CALL    $SAVAL                  ;; SAVE ALL REGISTERS
000440  062703  000034                  ADD     #I.PRM+10,R3            ;; POINT TO BUFFER DESCRIPTOR
000444  012700  000000G         READ::  MOV     #BUFFER,R0              ;; GET DESTINATION BUFFER ADDRESS
000450  000406                          BR      10$                     ;; JOIN COMMON CODE

000452                          RID::   CALL    $SAVAL                  ;; SAVE ALL REGISTERS
000456  062703  000026                  ADD     #I.PRM+2,R3             ;; POINT TO BUFFER DESCRIPTOR
000462  012700  000000G                 MOV     #IDBUFF,R0              ;; GET DESTINATION ID BUFFER ADDRESS

000466                          10$:    CALL    @RELOC                  ;; RELOCATE THE BUFFER          ; RJK02
000472  010146                          MOV     R1,-(SP)                ;; SAVE THE APR 6 BIAS
000474  010204                          MOV     R2,R4                   ;; COPY DESTINATION APR 6 DISPLACEMENT
000476  016300  000004                  MOV     4(R3),R0                ;; GET NUMBER OF BYTES TO MOVE
000502  011301                          MOV     (R3),R1                 ;; GET SOURCE APR 5 BIAS
000504  016302  000002                  MOV     2(R3),R2                ;; GET SOURCE APR 6 DISPLACEMENT
000510  042702  160000                  BIC     #160000,R2              ;; STRIP OFF APR OFFSET
000514  052702  120000                  BIS     #120000,R2              ;; MAKE SOURCE DISPLACEMENT VIA APR 5
000520  012603                          MOV     (SP)+,R3                ;; RETRIEVE DESTINTATION APR 6 BIAS
000522                                  CALLR   @BLXIO                  ;; COPY DATA AND RETURN         ; RJK02

                                        .DSABL  LSB
```

```
         000020                  MF$SPR = 20                       ; PRIORITY specified
         000040                  MF$CMK = 40                       ; CALL MASK specified
         000100                  MF$CVA = 100                      ; CALL VALUE specified
         000200                  MF$SNU = 200                      ; NUMBER specified
         000400                  MF$SAD = 400                      ; SUBADDRESS specified
         001000                  MF$SAC = 1000                     ; ACCOUNT specified
         002000                  MF$SND = 2000                     ; NODE specified
         004000                  MF$SPS = 4000                     ; PASSWORD specified
         010000                  MF$SUS = 10000                    ; USER specified
         020000                  MF$SGR = 20000                    ; GROUP specified
                                 ;
                                 ; Define flags in second parameter flags word (M$FLG2)
                                 ;
                                 ;
                                 ; Common parameters
                                 ;       NB: The common STATE and COUNTER TIMER parameters are kept
                                 ;       in M$FLG2 rather than M$FLAG to enable a simpler test of
                                 ;       the X25-SERVER and X25-PROTOCOL parameter groupings
                                 ;
         000001                  MF$STA = 1                        ; STATE CHANGE specified
         000002                  MF$PCT = 2                        ; COUNTER TIMER specified
                                 ;
                                 ; Module X25-PROTOCOL (GROUP)
                                 ;
         000010                  MF$GRO = 10                       ; GROUP specified (X25-PROTOCOL)
         000020                  MF$GNM = 20                       ; Group NUMBER specified
         000040                  MF$GTY = 40                       ; Group TYPE specified
         000100                  MF$GDT = 100                      ; Group DTE specified
                                 ;
                                 ; Module X25-PROTOCOL (DTE)
                                 ;
         001000                  MF$PDT = 1000                     ; DTE address specified (X25-PROTOCOL)
         002000                  MF$PCH = 2000                     ; DTE CHANNELS specified
         004000                  MF$PLN = 4000                     ; DTE LINE specified
         010000                  MF$PMC = 10000                    ; DTE MAX CIRCUITS specified
         020000                  MF$DNT = 20000                    ; Dummy flag to space-fill NETWORK field
                                 ;
                                 ; Define module entity flags word
                                 ;
         000001                  ME$XAC = 1                        ; Module X25-ACCESS
         000002                  ME$XPR = 2                        ; Module X25-PROTOCOL
         000003                  ME$X5S = 3                        ; Module X25-SERVER
         000004                  ME$X9S = 4                        ; Module X29-SERVER
                                 ;
                                 ; Define system-specific network management parameters for VMS compatibility
                                 ;
         005240                         MSVSUB = 2720.             ; DTE substate parameter
         000014                         MSVRUN = 12.               ; RUNNING substate
         000015                         MSVUNS = 13.               ; UNSYNCRHONISED substate
         005214                         MPVSTA = 2700.             ; X25-SERVER STATE parameter
                                 ;
      000000                            .PSECT
                                 ;
                                 ; TABLE OF LOGOUT ROUTINES (TERMINATED BY 0)
                                 ; ** NOTE ** THE NEXT FEW LINES ARE ORDER DEPENDENT
                                 ;
```

```
000122  001374                          BNE     60$                     ; IF NE, YES
000124  016700  000000G                 MOV     CURCTX,R0               ; GET THE ADDRESS OF THE CURRENT CONTEXT
000130  010160  000000G                 MOV     R1,C$SP(R0)             ; SAVE "SP"
000134  010067  000000G                 MOV     R0,LSTCTX               ; SET "LAST" CONTEXT ACCESSED
                                ;
                                ; SAVE "BUFFER" IN THE USER'S BUFFER
                                ;
000140                                  SWSTK$  80$                     ;; ENTER SYSTEM STATE
000144  016703  000000G                 MOV     CURIOP,R3               ;; GET THE I/O PACKET ADDRESS
000150  016301  000030                  MOV     I.PRM+4(R3),R1          ;; GET LENGTH OF USER'S BUFFER
000154  020167  177776G                 CMP     R1,BUFFER-2             ;; IS THE USER'S BUFF LARGER THAN BUFSIZ ?
000160  101402                          BLOS    70$                     ;; IF LOS, NO (OR EQUAL)
000162  016701  177776G                 MOV     BUFFER-2,R1             ;; ELSE, USE SMALLER SIZE
000166  010163  000030          70$:    MOV     R1,I.PRM+4(R3)          ;; TRUNCATE THE USER'S BUFFER IF NEEDED
000172                                  CALLR   WBUFF                   ;; "WRITE" THE BUFFER (TO SAVE IT)
000176                          80$:    CALLR   GETPKT                  ; AND TRY TO GET ANOTHER PACKET
```

```
000326  001376                   BNE     50$                     ; IF NE, MORE
000330  105712                   TSTB    (R2)                    ; WAS IT A RECOGNIZED ERROR ?
000332  001007                   BNE     60$                     ; IF NE, YES
000334  116701  177450           MOVB    NMSTAT,R1               ; ELSE PICK UP ERROR CODE
000340  012702  030412           MOV     #10.+400+<4000*6>,R2    ; SIGNED DECIMAL, NO LEADING ANYTHING
000344                           CALL    $CBTA                   ; CONVERT IT
000350  105020                   CLRB    (R0)+                   ; INSERT ZERO BYTE (BRIEFLY)
000352  005300           60$:    DEC     R0                      ; BACK OVER ZERO BYTE
000354  160300                   SUB     R3,R0                   ; CALCULATE MESSAGE LENGTH
000356  010067  177430           MOV     R0,NMSTAT+2             ; PASS BACK MESSAGE LENGTH
000362  162700  000004           SUB     #4,R0                   ; GET STRING LENGTH
000366  110063  000003           MOVB    R0,3(R3)                ; PLUG IT INTO MESSAGE
000372  000261           70$:    SEC                             ; INDICATE ERROR
000374  116700  177411   80$:    MOVB    NMSTAT+1,R0             ; SET RETURN CODE

                                 .IF DF  R$$PRO                  ; PRO/DECnet

                                 BMI     82$                     ; BRANCH IF UNSUCCESSFUL OR DONE RETURN CODE
                                 BIT     #140,R0                 ; TEST NT.CFG AND ALIAS FILES UPDATE BITS
                                 BEQ     88$                     ; IF EQ, DON'T UPDATE FILE
                                 BIC     #140,R0                 ; ELSE CLEAR NT.CFG AND ALIAS FILES UPDATE BITS
                                 MOV     #1,FILUPD               ; AND SET FILE UPDATE FLAG
                                 BR      88$                     ; CONTINUE

                         82$:    CMPB    #MS.DON,R0              ; IS REQUEST DONE ?
                                 BNE     88$                     ; IF NE, UNSUCCESSFUL - BRANCH
                                 TST     FILUPD                  ; TEST FOR CONFIGURATION FILE UPDATE
                                 BEQ     88$                     ; IF EQ, DON'T UPDATE FILE
                                 CLR     FILUPD                  ; ELSE CLEAR FILE UPDATE FLAG
                                 MOV     R0,-(SP)                ; SAVE R0
                                 MOV     NMSTAT+2,-(SP)          ; SAVE 2ND WORD OF I/O STATUS BLOCK

                         ;
                         ; Update the approoriate files by issuing I/O request
                         ;
                                 QIOW$S  #NM.INI,#$TMLUN,#$TMEFN,,#NMSTAT,,<#FILEUP,,,#NMFBUF,,#NMFMES>
                                 BCC     83$                     ; IF CC, DIRECTIVE IS ALRIGHT
                                 CLRB    NMSTAT+1                ; ZERO TO FALL INTO ERROR PROCESSING

                         83$:    TSTB    NMSTAT+1                ; CHECK NICE RETURN CODE FROM NMFACP
                                 BGT     85$                     ; IF GT, FILE WAS UPDATED CORRECTLY
                                 CMP     (SP)+,(SP)+             ; POP NMSTAT+2 AND R0 FROM STACK
                                 MOV     #IE.RSU,NMSTAT          ; SET DSW ERROR CODE FOR ERROR TABLE
                                                                 ; (NOTE: THIS CODE SHOULD NOT COME BACK
                                                                 ;  FROM A QIO$ OR MATCH ANY OTHER ERROR
                                                                 ;  TABLE CODES FOR THE DSW.)
                                 BR      25$                     ; BRANCH TO BUILD ERROR RETURN
                         85$:    MOV     (SP)+,NMSTAT+2          ; RESTORE 2ND WORD OF I/O STATUS BLOCK
                                 MOV     (SP)+,R0                ; RESTORE R0

                                 .ENDC   ; DF R$$PRO

000400  110013           88$:    MOVB    R0,(R3)                 ; PUT NICE RETURN CODE IN BUFFER

                                 .IF DF  R$$PRO                  ; PRO/DECNET

                                 BPL     89$                     ; IF PL, CONTINUE
```

```
                                                                   N 10
MANQIO     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31      PAGE 4
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME      REFERENCES

ALUN$           #5-55      5-92
CALL            6-188
CALLR           #5-55      7-408      7-438
DIR$            #5-55      6-156     #6-158      6-158     #6-161      6-161
ERROR           #5-76      5-106      5-107      5-108      5-109      5-110      5-111      5-112      5-113      5-114
                5-115      5-123
MANDF$          #5-55       5-57
MOV$           #6-158      6-158      6-158      6-158      6-158      6-158      6-158     #6-158      6-158      6-158
               #6-158      6-158      6-158     #6-161      6-161      6-161     #6-161      6-161      6-161     #6-161
                6-161      6-161
MVB$           #6-158      6-158     #6-161      6-161
OFF$            #5-92
QDPB$S         #6-158      6-158     #6-161      6-161
QIOW$           #5-55
QIOW$S          #5-55      6-158      6-161
RETURN          6-249
RVP$           #6-158      6-158      6-158      6-158     #6-161      6-161      6-161      6-161
```

```
MANTRA     CREATED BY  MACRO  ON 29-JUN-85 AT 10:31      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

A.ACC      000020           #5-57       9-198
A.LNK      000000           #5-57
A.NAM      000002           #5-57
A.PUD      000010           #5-57
A.REM      000012           #5-57
A.UCB      000010           #5-57
BIAS     = 000002           #5-60      *9-207
CURIOP   = ******  GX         6-83      7-116       8-149       9-184
I.PRM    = ******  GX         6-84      7-117       8-150       9-185
KISAR6   = ******  GX         9-192    *9-213
MAPADD   = ******  GX         6-90
MAPALI   = ******  GX         8-155
MAPREM   = ******  GX         7-122
ME.FOR     177776             6-86      7-119       8-152       9-187
ME.UCO     177770             6-92      7-124       8-157       9-217
MS.SUC     000001             6-94      7-126       8-162       9-229
M$$MGE   = 000000             5-59
N$$VCT   = ******             9-192     9-195       9-213
REPALI   = ******  GX         9-193
TRAACC     000216 RG        #9-182
TRAADD     000000 RG        #6-81
TRAALI     000134 RG        #8-147
TRANAM     000056 RG        #7-114
$HEADR   = ******  GX         9-195
```

```
                                                                       N 12
**FILE**ID**MANVTB




MM      MM    AAAAAA    NN      NN  VV      VV  TTTTTTTTTT  BBBBBBBB
MM      MM    AAAAAA    NN      NN  VV      VV  TTTTTTTTTT  BBBBBBBB
MMMM  MMMM  AA      AA  NN      NN  VV      VV      TT      BB      BB
MMMM  MMMM  AA      AA  NN      NN  VV      VV      TT      BB      BB
MM  MM  MM  AA      AA  NNNN    NN  VV      VV      TT      BB      BB
MM  MM  MM  AA      AA  NNNN    NN  VV      VV      TT      BB      BB
MM      MM  AA      AA  NN  NN  NN  VV      VV      TT      BBBBBBBB
MM      MM  AA      AA  NN  NN  NN  VV      VV      TT      BBBBBBBB
MM      MM  AAAAAAAAAA  NN    NNNN  VV      VV      TT      BB      BB
MM      MM  AAAAAAAAAA  NN    NNNN  VV      VV      TT      BB      BB
MM      MM  AA      AA  NN      NN    VV  VV        TT      BB      BB        ....
MM      MM  AA      AA  NN      NN    VV  VV        TT      BB      BB        ....
MM      MM  AA      AA  NN      NN      VV          TT      BBBBBBBB          ....
MM      MM  AA      AA  NN      NN      VV          TT      BBBBBBBB          ....




LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

N 13

```
**FILE**ID**PSIBYE





PPPPPPPP       SSSSSSSS      IIIIII     BBBBBBBB     YY      YY   EEEEEEEEEE
PPPPPPPP       SSSSSSSS      IIIIII     BBBBBBBB     YY      YY   EEEEEEEEEE
PP      PP   SS                II       BB      BB   YY      YY   EE
PP      PP   SS                II       BB      BB   YY      YY   EE
PP      PP   SS                II       BB      BB     YY  YY     EE
PP      PP   SS                II       BB      BB     YY  YY     EE
PPPPPPPP       SSSSSS          II       BBBBBBBB         YY       EEEEEEEE
PPPPPPPP       SSSSSS          II       BBBBBBBB         YY       EEEEEEEE
PP                   SS        II       BB      BB       YY       EE
PP                   SS        II       BB      BB       YY       EE
PP                   SS        II       BB      BB       YY       EE                ....
PP                   SS        II       BB      BB       YY       EE                ....
PP           SSSSSSSS        IIIIII     BBBBBBBB         YY       EEEEEEEEEE        ....
PP           SSSSSSSS        IIIIII     BBBBBBBB         YY       EEEEEEEEEE        ....




LL             SSSSSSSS    TTTTTTTTTT
LL             SSSSSSSS    TTTTTTTTTT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL             SSSSSS          TT
LL             SSSSSS          TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LLLLLLLLLL   SSSSSSSS          TT
LLLLLLLLLL   SSSSSSSS          TT
```

```
                                                    .SBTTL  ALCHIN - ALIAS CHANGE PARAMETERS INITIALIZE
                                            ;+
                                            ; **-ALCHIN-ALIAS CHANGE PARAMETERS INITIALIZE ROUTINE
                                            ;
                                            ; THIS ROUTINE IS CALLED TO INITIALIZE FOR THE ALIAS CHANGE PARAMETER
                                            ; OPERATION.
                                            ;
                                            ; INPUTS:
                                            ;       R0 = OPTION (SET/CLEAR)
                                            ;       R1 = ID ADDRESS (ALIAS FORMAT)
                                            ;       R2 = LENGTH OF DATA IN INPUT BUFFER
                                            ;       R3 = ADDRESS OF INPUT BUFFER
                                            ;       R4 = ADDRESS OF CONTEXT AREA
                                            ;
                                            ; OUTPUTS:
                                            ;       R0 = RETURN CODE
                                            ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                            ;       R3 = ADDESS OF OUTPUT BUFFER
                                            ;       DATA SAVED IN CONTEXT AREA
                                            ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                            ;
                                            ; REGISTERS:
                                            ;       R3, R4, R5 ARE PRESERVED
                                            ;-

000000                                      ALCHIN::
000000  004567  000000G                             JSR     R5,$SAVRG               ; SAVE R3-R5
000004  010064  000002                              MOV     R0,A$OPT(R4)            ; SAVE OPTION WORD
000010  010364  000006                              MOV     R3,A$BUF(R4)            ; SAVE BUFFER POINTER
                                            ;
                                            ; CHECK LENGTH OF CONTEXT AREA
                                            ;
000014  012700  177773                              MOV     #ME.MPR,R0              ; ASSUME CONTEXT AREA TOO SMALL
000020  026427  177776  000103                      CMP     -2(R4),#A$LEN           ; IS CONTEXT AREA LARGE ENOUGH ?
000026  002474                                      BLT     30$                     ; IF LT, NO - ERROR
                                            ;
                                            ; CHECK OPTIONS FLAGS
                                            ;
000030  012700  177777                              MOV     #ME.FUN,R0              ; ASSUME ILLEGAL OPTION
000034  032764  177677  000002                      BIT     #^C<MO$CLE>,A$OPT(R4)   ; CHECK FOR ILLEGAL OPTIONS
000042  001066                                      BNE     30$                     ; IF NE, ERROR
                                            ;
                                            ; CHECK ALIAS FORMAT
                                            ;
000044                                              CALL    CHKALI                  ; CHECK ALIAS FORMAT
000050  103463                                      BCS     30$                     ; IF CS, ILLEGAL FORMAT
                                            ;
                                            ; CHECK SCOPE, UNLESS OPERATION WAS FOR "ALL"
                                            ;
000052  126427  000011  177775                      CMPB    A$TYP(R4),#MF$ALL       ; WAS ALIAS TYPE "ALL"
000060  001005                                      BNE     4$                      ; IF NE, NO - CHECK SCOPE
000062  032764  000100  000002                      BIT     #MO$CLE,A$OPT(R4)       ; ELSE, WAS THIS A CLEAR OPERATION ?
000070  001045                                      BNE     20$                     ; IF NE, YES - BYPASS DESTINATION CHECK
000072  000452                                      BR      30$                     ; ELSE, ILLEGAL ALIAS TYPE FOR SET
000074                                      4$:     CALL    CHKSCO                  ; CHECK THE SCOPE
000100  103447                                      BCS     30$                     ; IF CS, ILLEGAL SCOPE
                                            ;
```

```
     53                                    .SBTTL  MACRO CALLS AND LOCAL DATA
     54                               ;
     55                               ; MACRO LIBRARY CALLS
     56                               ;
     57                                    .MCALL  ANBDF$,MANDF$,SWSTK$,SAVMAP,MAP,RESMAP,SAVRG,RESRG
     58                                    .MCALL  HDRDF$,CALLX,DHBDF$,ALCCX$
     59
     60 000000                             ANBDF$                          ; DEFINE ALIAS NAME BLOCK OFFSETS
     61 000000                             MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     62 000000                             HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     63 000000                             DHBDF$                          ; DEFINE DEC HOME BLOCK OFFSETS
     64
     65          000000                    BIAS=0                          ; BIAS OF REGISTERS ON STACK
     66                                    .IF DF  M$$MGE
     67          000002                    BIAS=2                          ; OFFSET FOR SAVED KISAR6
     68                                    .ENDC
     69
     70 000000                             ALCCX$  LIST                    ; DEFINE CONTEXT AREA OFFSETS
                                      ;
                                      ; CONTEXT AREA OFFSET DEFINITIONS
                                      ;
         000020                            MAXID=16.                       ; MAXIMUM LENGTH OF USER ID
         000010                            MAXPAS=8.                       ; MAXIMUM LENGTH OF PASSWORD
         000020                            MAXACC=16.                      ; MAXIMUM LENGTH OF ACCOUNT
    000020                                 .ASECT
         000000                            .=0
                                      ;
                                      ; OFFSETS IN COMMON WITH READ INFORMATION MODULES
                                      ;
    000000                            A$FLG:  .BLKW   1                      ; FLAG WORD
    000002                            A$OPT:  .BLKW   1                      ; SAVED OPTIONS WORD
    000004                            A$PAR:  .BLKW   1                      ; CURRENT PARAMETER TYPE
    000006                            A$BUF:  .BLKW   1                      ; SAVED POINTER TO BUFFER
    000010                            A$NUM:  .BLKB   1                      ; NUMBER OF BYTES IN ANB (FOR SET)
    000011                            A$TYP:  .BLKB   1                      ; ALIAS FORMAT TYPE
    000012                            A$ALI:  .BLKW   3                      ; ALIAS NAME, SPACE FILLED
    000020                            A$SCO:  .BLKW   1                      ; SCOPE OF ALIAS
                                      ;
                                      ; OFFSETS UNIQUE TO CHANGE PARAMETERS MODULES
                                      ;
    000022                            A$DES:  .BLKW   3                      ; DESTINATION NAME, SPACE FILLED
    000030                                    .BLKB   <1+MAXID>+<1+MAXPAS>+<1+MAXACC> ; ACCESS CONTROL STORAGE
    000103                            A$LEN:                                 ; LENGTH OF CONTEXT AREA
    000000                                    .PSECT
                                      ;
                                      ; FLAGS WORD BIT DEFINITIONS
                                      ;
         000001                               AF.INI=1                       ; SUCCESSFUL INITIALIZATION COMPLETED
         000002                               AF.OVR=2                       ; CONTEXT AREA OVERFLOWED
     71
```

B	1	
C	1	
D	1	
E	1	
F	1	
G	1	
H	1	
I	1	
J	1	
K	1	
L	1	
M	1	
N	1	
B	2	
C	2	
D	2	
E	2	
F	2	
G	2	
H	2	
I	2	
J	2	
K	2	
L	2	
M	2	
N	2	
B	3	
C	3	MANACP
D	3	
E	3	
F	3	
G	3	
H	3	
I	3	
J	3	
K	3	
L	3	
M	3	
N	3	
B	4	
C	4	
D	4	
E	4	
F	4	
G	4	
H	4	
I	4	
J	4	
K	4	
L	4	
M	4	
N	4	
B	5	
C	5	
D	5	
E	5	
F	5	
G	5	MANAST
H	5	
I	5	
J	5	
K	5	
L	5	
M	5	
N	5	
B	6	
C	6	
D	6	MANBUF
E	6	
F	6	
G	6	
H	6	
I	6	
J	6	
K	6	
L	6	
M	6	
N	6	
B	7	
C	7	
D	7	
E	7	
F	7	
G	7	
H	7	MANBYE
I	7	
J	7	
K	7	
L	7	
M	7	
N	7	
B	8	
C	8	
D	8	
E	8	
F	8	
G	8	
H	8	
I	8	MANFRK
J	8	
K	8	
L	8	
M	8	
N	8	
B	9	
C	9	
D	9	
E	9	
F	9	
G	9	
H	9	MANQIO
I	9	
J	9	
K	9	
L	9	
M	9	
N	9	
B	10	
C	10	
D	10	
E	10	
F	10	
G	10	
H	10	
I	10	
J	10	
K	10	
L	10	
M	10	
N	10	
B	11	MANTRA
C	11	
D	11	
E	11	
F	11	
G	11	
H	11	
I	11	
J	11	
K	11	
L	11	
M	11	
N	11	
B	12	
C	12	MANVCD
D	12	
E	12	
F	12	
G	12	
H	12	
I	12	
J	12	
K	12	
L	12	
M	12	
N	12	MANVTB
B	13	
C	13	
D	13	
E	13	
F	13	
G	13	
H	13	
I	13	
J	13	
K	13	
L	13	
M	13	
N	13	PSIBYE
B	14	
C	14	
D	14	
E	14	
F	14	
G	14	
H	14	
I	14	
J	14	ALCHIN
K	14	
L	14	
M	14	
N	14	
B	15	
C	15	
D	15	
E	15	
F	15	
G	15	
H	15	
I	15	
J	15	
K	15	ALCHOP
L	15	
M	15	
N	15	
B	16	
C	16	
D	16	
E	16	
F	16	
G	16	
H	16	
I	16	
J	16	
K	16	
L	16	
M	16	

```
                                                                   B  1
ALCHOP      CREATED BY  MACRO  ON 10-SEP-85 AT 11:46     PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE           REFERENCES

AF.INI  = 000001        #5-70      6-98
AF.OVR  = 000002        #5-70
ALCHOP    000000 RG     #6-94
ALOCB   = ******  GX     7-195
A$ALI     000012        #5-70      7-217     10-437     10-439     10-441
A$BUF     000006        #5-70     *6-96      6-113
A$DES     000022        #5-70
A$FLG     000000        #5-70      6-98     *6-111
A$LEN     000103        #5-70
A$NUM     000010        #5-70      7-190
A$OPT     000002        #5-70      6-103
A$PAR     000004        #5-70     *8-316    *8-320
A$SCO     000020        #5-70      7-168     7-243      7-252      8-336      10-443
A$TYP     000011        #5-70      8-308     8-334
A.ACC     000020        #5-60      9-384
A.LNK     000000        #5-60
A.NAM     000002        #5-60      10-437    10-439     10-441
A.PUD     000010        #5-60
A.REM     000012        #5-60
A.UCB     000010        #5-60      7-168     7-241      7-252      8-336      10-443
BIAS    = 000002        #5-65     #5-67     *7-197     *8-315
CEACC   = ******  GX     7-214     7-238     8-331      9-399      10-433
CF$VR2  = ******  GX     8-318
CLRALI    000336 R       6-105    #8-290
CURCTX  = ******  GX     8-317
C$FLAG  = ******  GX     8-318
DEAANB    000534 R       7-170     8-322     8-338     #9-373
DEACB   = ******  GX     9-396
DECPT   = ******  GX     7-160     7-225     8-305
D$ANN     000000         7-162     7-233     8-307
FINDA     000634 R       7-166     8-313    #10-428
KISAR6  = ******  GX     7-159    *7-256     8-304     *8-341
MAXACC  = 000020        #5-70      5-70
MAXID   = 000020        #5-70      5-70
MAXPAS  = 000010        #5-70      5-70
ME$ALI    000010         8-316
ME$AL2    000003         8-320
ME.DON    177600         6-97
ME.RES    177761         7-197
ME.UCO    177770         8-315
MF$ALL    177775         8-334
MO$CLE    000100         6-103
MS.SUC    000001         7-146     8-291
M$$MGE  = 000000         5-66
N$$VCT  = ******         7-159     7-249     7-256      8-304      8-341      9-395
R$$PRO  = ******         7-148     7-178     7-199      7-227      7-258      8-293      8-343      9-375
SETALI    000072 R       6-107    #7-145
$CALLX  = ******  GX     7-249     9-395
$CRUNM  = ******  GX     7-172    *7-234     7-235     *7-236     *8-327      8-328     *8-329      9-392     *9-397
                         9-398    *10-429    10-430    *10-431

$PRUNM  = ******  GX    *7-235     7-248    *8-328      9-393      9-397     *10-430
$SAVRG  = ******  GX     6-95
```

```
                                                                   C  1
HOP         CREATED BY  MACRO  ON 10-SEP-85 AT 11:46     PAGE 2
SYMBOL CROSS REFERENCE                                   CREF    04.00
```

```
                                                      B  2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-1
Symbol table

AF.INI= 000001            D$NA    000062            I$$RDN= 000000            MC$MBL  001764            MC$XLR  002330
AF.OVR= 000002            D$NBEA  000056            KISAR6= ****** GX         MC$MBX  004420            MC$XMA  000310
ALREIN  000000RG          D$NBRA  000054            K$$CNT= 177546            MC$MBY  001752            MC$XMC  002305
A$ALI   000012            D$NEND= 000054            K$$CSR= 177546            MC$MPX  004406            MC$XMS  002304
A$BUF   000006            D$NLN   000030            K$$LDC= 000000            MC$NAP  001604            MC$XNR  002332
A$FLG   000000            D$NN    000060            K$$TPS= 000074            MC$NBR  001130            MC$XRC  002316
A$LEN   000024            D$OUTT  000043            LC$EXT  000000            MC$NBS  001131            MC$XRR  002331
A$NEX   000022            D$RETF  000050            LC$INT  000001            MC$NCR  001154            MC$XRS  002342
A$NUM   000010            D$RNN   000002            LD$LF = 000000            MC$NCS  001155            MC$020  000002
A$OPT   000002            D$RTMR  000076            L$$ASG= 000000            MC$NML  001274            MD$CI   000007
A$PAR   000004            D$SEG   000036            L$$DRV= 000000            MC$NMR  001142            MD$CNA  000003
A$SAV   000024            D$SER   000032            L$$P11= 000001            MC$NMS  001143            MD$COU  000001
A$SCO   000020            D$SQRL  000052            L$$11R= 000000            MC$NNO  001606            MD$DA   000010
A$TYP   000011            D$$BUG= 177514            MAXACC= 000020            MC$NNU  001605            MD$DL   000004
A$$CHK= 000000            D$$ISK= 000000            MAXID = 000020            MC$NDP  001607            MD$DLV  000020
A$$CPS= 000000            D$$L11= 000001            MAXPAS= 000010            MC$NOR  000000            MD$DMC  000014
A$$PRI= 000000            D$$YNC= 000000            MB$DIS  000001            MC$NPF  001616            MD$DMF  000046
A$$TRP= 000000            D$$YNM= 000000            MB$ENA  000000            MC$NPR  001630            MD$DMP  000022
A.ACC   000020            E$$XPR= 000000            MB$FUL  000002            MC$NRE  001200            MD$DMR  000050
A.LNK   000000            FINDA   000466R           MB$MIX  000002            MC$NRT  001166            MD$DMV  000042
A.NAM   000002            F$$LVL= 000001            MB$ONE  000001            MC$NVR  001642            MD$DN   000016
A.PUD   000010            G$$TPP= 000000            MB$RXD  000001            MC$P08  000000            MD$DP   000000
A.REM   000012            G$$TSS= 000000            MB$TXO  000000            MC$P11  000001            MD$DPV  000044
A.UCB   000010            G$$TTK= 000000            MB$ZER  000000            MC$RFL  002046            MD$DQ   000006
CEACC = ****** GX         G$$WRD= 000000            MC$BAB  006073            MC$SBU  002051            MD$DTE  000024
CF$VR2= ****** GX         H.CSP   000000            MC$BID  001765            MC$SFL  002044            MD$DU   000002
CHKALI= ****** GX         H.CUIC  000010            MC$BMC  001767            MC$SYC  000012            MD$DUP  000012
CHKSCO= ****** GX         H.DSW   000046            MC$BSC  001766            MC$SYL  000036            MD$DV   000026
CURCTX= ****** GX         H.DUIC  000012            MC$CAC  001442            MC$SYR  000050            MD$DZ   000030
C$FLAG= ****** GX         H.EFLM  000004            MC$CAP  001440            MC$SYS  000024            MD$FUL  000000
C$$CKP= 000000            H.EFSV  000040            MC$CCL  001445            MC$SYZ  000000            MD$HAL  000001
C$$DRE= 000400            H.FCS   000050            MC$CDC  002045            MC$TBR  001140            MD$HEL  000002
C$$RSH= 177564            H.FORT  000052            MC$CDP  001441            MC$TBS  001141            MD$KCP  000013
DECPT = ****** GX         H.FPSA  000042            MC$CIF  001465            MC$TDR  002053            MD$KDP  000034
D$AMXC  000072            H.FPSL= 000062            MC$CLD  001464            MC$TIM  000000            MD$KDZ  000036
D$AMXH  000074            H.FPVA  000034            MC$CDU  100000            MC$TYP  007777            MD$KL   000040
D$ANN   000000            H.GARD  000072            MC$CTL  001454            MC$UBU  002052            MD$KMX  000054
D$BRPR  000102            H.HDLN  000002            MC$CTR  001452            MC$UFD  002047            MD$KMY  000052
D$BRTM  000100            H.IPC   000016            MC$CTS  001453            MC$UMR  001132            MD$LEN  000050
D$DELF  000045            H.IPS   000014            MC$DDV  002050            MC$UMS  001133            MD$PCL  000011
D$DELW  000046            H.ISP   000020            MC$LBR  001750            MC$VAX  000003            MD$QNA  000005
D$END = 000104            H.LUN   000076            MC$LBS  001751            MC$WID  060000            MD$UNA  000001
D$FNB   000034            H.NLUN  000074            MC$LDI  001774            MC$WIH  040000            MD$WIT  000002
D$HIDR  000024            H.NML   000061            MC$LDD  001775            MC$WIL  020000            MD.ABO  000017
D$HOST  000022            H.ODVA  000022            MC$LDR  001762            MC$W08  020000            MD.ACC  000010
D$INAC  000044            H.ODVL  000024            MC$LDS  001763            MC$W16  040000            MD.AOB  000016
D$INCT  000042            H.OVLY  000054            MC$LLB  002021            MC$W32  060000            MD.BOB  000011
D$IPL   000051            H.PFVA  000032            MC$LLP  002115            MC$XBR  001750            MD.DIA  000007
D$LID   000020            H.RCVA  000036            MC$LLR  002007            MC$XBS  001751            MD.DOB  000015
D$LNAM  000006            H.RRVA  000062            MC$LOO  000001            MC$XCJ  000322            MD.DUM  000002
D$LNUM  000014            H.SPRI  000060            MC$LRB  002020            MC$XCR  002260            MD.FAI  000014
D$LST   000047            H.TKVA  000026            MC$LRP  002114            MC$XCS  002261            MD.FNA  000001
D$MAXC  000064            H.TKVL  000030            MC$LRR  002006            MC$XDR  001762            MD.FDB  000006
D$MAXH  000066            H.VEXT  000056            MC$LSI  002032            MC$XDS  001763            MD.LDA  000001
D$MAXV  000070            H.WND   000044            MC$LST  002033            MC$XFR  002272            MD.LSH  000020
D$MLL   000040            H.X25   000064            MC$LUP  006072            MC$XFS  002273            MD.NOB  000012
D$MNOD  000041            I$$RAR= 000000            MC$MAP  010000            MC$XLJ  000323            MD.NON  177777
```

```
                                                      C  2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-2
Symbol table

MD.PER  000000            ME$ADD  000000            MO$PRS  000001            MP$CPU  000161            MP$LLE  000227
```

```
                                               .SBTTL  FMTNAM - FORMAT A NAME FIELD
                                       ;+
                                       ; **-FMTNAM-FORMAT A NAME FIELD
                                       ;
                                       ; THIS ROUTINE TAKES A 6 CHARACTER SPACE FILLED NODE NAME, AND
                                       ; BUILDS AN IMAGE FIELD, DELETING TRAILING SPACES.
                                       ;
                                       ; INPUTS:
                                       ;       R1 = POINTER TO 6 CHARACTER NAME
                                       ;       R3 = POINTER TO BUFFER FOR IMAGE STRING
                                       ;
                                       ; OUTPUTS:
                                       ;       R1 = NEXT BYTE AFTER NAME
                                       ;       R3 = NEXT FREE BYTE IN BUFFER
                                       ;
                                       ; REGISTERS:
                                       ;       R0, R4 ARE PRESERVED
                                       ;
                                       ;-

 000306                                FMTNAM:
 000306  010302                                MOV     R3,R2                   ; COPY BUFFER POINTER
 000310  012705  000006                        MOV     #6,R5                   ; SET LENGTH OF NAME
 000314  110523                                MOVB    R5,(R3)+                ; ASSUME IMAGE FIELD LENGTH OF 6
 000316  112123                        10$:    MOVB    (R1)+,(R3)+             ; MOVE NAME INTO BUFFER
 000320  005305                                DEC     R5                      ; ...
 000322  001375                                BNE     10$                     ; ...
 000324  126227  000001  000040                CMPB    1(R2),#'                ; IS THE FIRST CHARACTER A SPACE ?
 000332  001407                                BEQ     30$                     ; IF EQ, YES - DON'T DELETE SPACES
 000334  126327  177777  000040        20$:    CMPB    -1(R3),#'               ; IS THE LAST CHARACTER A SPACE ?
 000342  001003                                BNE     30$                     ; IF NE, NO - FINISHED
 000344  005303                                DEC     R3                      ; ELSE, BACK UP OVER SPACE
 000346  105312                                DECB    (R2)                    ; DEC IMAGE FIELD LENGTH
 000350  000771                                BR      20$                     ; AND LOOK AT PREVIOUS CHAR
 000352                                30$:    RETURN
```

```
                                               .SBTTL  FMTSCO - FORMAT SCOPE PARAMETER
```

```
                .TITLE  ALXXIN - ALIAS INITIALIZATION COMMON ROUTINES
                .IDENT  /V05.00/
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - ALIAS INITIALIZATION COMMON ROUTINES
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
;
```

```
                .SBTTL  MACRO CALLS AND LOCAL DATA
```

```
                                                                       B  5
ALXXIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07      PAGE  1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE            REFERENCES

AF.INI  = 000001         #5-65
AF.OVR  = 000002         #5-65
ALIIDY    000026 R       #5-84         6-138
ALIT      000000 R       #5-74         5-84
ALIT2     000010 R       #5-77         5-85
ALIT3     000016 R       #5-80         5-86
A$ALI     000012         #5-65         6-125
A$BUF     000006         #5-65
A$DES     000022         #5-65
A$FLG     000000         #5-65
A$LEN     000103         #5-65
A$NUM     000010         #5-65
A$OPT     000002         #5-65
A$PAR     000004         #5-65        *6-149       *6-153       *7-216
A$SCO     000020         #5-65        *7-222       *7-266       *7-275        7-285       *7-317
A$TYP     000011         #5-65        *6-119
A.ACC     000020         #5-60
A.LNK     000000         #5-60
A.NAM     000002         #5-60
A.PUD     000010         #5-60
A.REM     000012         #5-60
A.UCB     000010         #5-60
CF$VR2  = ******   GX     6-151
CHKALI    000036 RG      #6-112
CHKPAR    000526 RG       7-217       #8-347
CHKSCO    000252 RG      #7-211
CMPCST    000210 R        6-141       #6-171
CURCTX  = ******   GX     6-150
C$FLAG  = ******   GX     6-151
DEALUN  = ******   GX     7-283
DV.TTY  = 000004          7-314
D.NAM   = ******   GX     7-310
HEADR   = ******   GX     7-300
H.LUN     000076          7-301
MAXACC  = 000020         #5-65         5-65
MAXID   = 000020         #5-65         5-65
MAXPAS  = 000010         #5-65         5-65
ME$ALI    000010          6-149
ME$AL2    000003          6-153
ME.PMI    177743          8-350
ME.PTY    177772          8-355
ME.PVA    177760          7-321
ME.UCO    177770          6-154
MF$ALL    177775          6-117
MP$SCO    000144          7-215
MS$TER    000001          7-247
N$$VCT  = ******          7-316
R$$MPL  = ******          7-290
R$$PRO  = ******          7-225        7-237        7-259        7-268
SAVNAM  = ******   GX     6-127
TKTCB   = ******   GX     7-312
T.UCB   = ******   GX     7-313
```

```
                                                                       C  5
ALXXIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07      PAGE  2
SYMBOL CROSS REFERENCE                                   CREF    04.00
```

```
S1.TSY= 000040            S5.HPC= 000014            US.VV = 000001            U.SHUN= 000074            U3.OPA= 100000
S1.USI= 100000            S5.HPO= 000020            US.WCK= 000010            U.SPC = 000036            U3.PAR= 040000
S2.BEL= 020000            S5.ITI= 000100            UU.ABO= 000400            U.STS   000005            U3.UPC= 020000
S2.BRQ= 000020            S5.OXF= 000040            UU.ATN= 000100            U.ST2   000007            U4.CR = 000100
S2.CR = 002000            S5.RPO= 002000            UU.AVN= 000004            U.SUB = 000036            V$$CTR= 001000
S2.CTO= 040000            S5.SW1= 000001            UU.GUS= 000010            U.TCHP  000042            W.BATT  000006
S2.CTS= 100000            S5.TMM= 000002            UU.IOS= 002000            U.TCVP  000043            W.BFPD  000014
S2.ELF= 001000            S5.VER= 010000            UU.ONL= 000020            U.TFLK  000040            W.BHVR  000004
S2.FLF= 000400            S5.XOF= 000004            UU.RCT= 000002            U.TFRQ  000036            W.BLGH  000020
S2.IRQ= 000200            S5.XON= 000010            UU.RDY= 000200            U.TIXL  000060            W.BLPD  000016
S2.OBF= 004000            S6.EIO= 000400            UU.SER= 000001            U.TLPP  000053            W.BLVR  000002
S2.ORQ= 000100            S6.RDI= 100000            UU.SIO= 001000            U.TMTI  000047            W.BNPD  000015
S2.PCU= 010000            S6.RLU= 001000            UU.SPC= 000040            U.TRCK= 000100            W.BOFF  000012
S2.RCU= 000001            T$$KMG= 000000            U.ACB   000062            U.TSTA  000026            W.BPCB  000000
S2.SRQ= 000040            T$$MIN= 000000            U.ACP = 000032            U.TST5  000054            W.BSIZ  000010
S2.WAL= 000010            UC.ALG= 000200            U.ADMA  000066            U.TST6  000056            X$$DBT= 000000
S2.WRA= 000006            UC.ATT= 000010            U.AFLG  000064            U.TTAB  000050            ZF.COU= 001000
S2.WRB= 000002            UC.KIL= 000004            U.ATT   000022            U.TTYP  000046            ZF.DDM= 000001
S3.ACR= 000001            UC.LGH= 000003            U.BPKT= 000044            U.TUX   000024            ZF.DIA= 004000
S3.CTC= 000004            UC.NPR= 000100            U.BUF   000024            U.UHVR= 000107            ZF.DLC= 000002
S3.FDX= 000200            UC.PWF= 000020            U.CBF = 000032            U.UIC   000044            ZF.DVP= 100000
S3.ICE= 001000            UC.QUE= 000040            U.CLI   177772            U.UNFL= 000052            ZF.INI= 040000
S3.MHE= 000400            UD.UNS= 000000            U.CNT   000030            U.UNIT  000006            ZF.KMX= 000020
S3.NEC= 000020            UD.160= 000004            U.COTQ  000030            U.UNSZ= 000114            ZF.LLC= 000004
S3.PPT= 020000            UD.200= 000001            U.CTCB  000026            U.UNTI= 000060            ZF.LMC= 000100
S3.PTH= 004000            UD.556= 000002            U.CTL   000004            U.USVR= 000106            ZF.MAN= 020000
S3.RAL= 000010            UD.625= 000005            U.CTYP  000052            U.UTIL= 000036            ZF.MFL= 000010
S3.RES= 010000            UD.8K = 000006            U.CW1   000010            U.VCB = 000034            ZF.MTM= 000400
S3.RUB= 040000            UD.800= 000003            U.CW2   000012            U.VSER= 000120            ZF.MUX= 000040
S3.TAB= 000002            UM.CLI= 000036            U.CW3   000014            U.2MED= 000070            ZF.PSE= 002000
S3.TME= 002000            UM.DSB= 000200            U.CW4   000016            U2.AT.= 000020            ZF.SLI= 010000
S3.TSY= 000040            UM.NBR= 000400            U.CYL = 000104            U2.CRT= 002000            ZF.TIM= 000200
S3.8BC= 000100            UM.OVR= 000001            U.DCB   000000            U2.DH1= 100000            ZF.X3P= 000000
S4.ABD= 100000            US.ABO= 000001            U.FCDE= 000042            U2.DJ1= 040000            ZS.ASN= 100000
S4.ANI= 000400            US.BSP= 000002            U.FNUM= 000040            U2.DZ1= 000100            ZS.BSY= 140000
S4.AVO= 001000            US.BSY= 000200            U.GRP = 000102            U2.ESC= 001000            Z.AVL   000014
S4.BLK= 002000            US.FOR= 000040            U.KCSR= 000032            U2.HFF= 010000            Z.DAT   000016
S4.DEC= 004000            US.FRK= 000002            U.KCS6= 000034            U2.HLD= 000040            Z.DSP   000000
S4.DLO= 000100            US.KPF= 000001            U.LUIC  177774            U2.LOG= 000400            Z.FLG   000010
S4.EDT= 010000            US.LAB= 000004            U.MEDI= 000040            U2.LWC= 000001            Z.LEN = 000016
S4.HFF= 000020            US.MDE= 000002            U.MLUN= 000050            U2.L3S= 000004            Z.LLN   000006
S4.HFL= 000007            US.MDM= 000020            U.MUP   177772            U2.L8S= 010000            Z.MAP   000020
S4.HHT= 000040            US.MNT= 000100            U.OWN   177776            U2.NEC= 004000            Z.NAM   000004
S4.HSY= 000200            US.OFL= 000001            U.RBNS= 000112            U2.PRV= 000010            Z.PCB   000012
S4.RGS= 020000            US.PUB= 000004            U.RCTC= 000113            U2.RMT= 020000            Z.SCH   000007
S4.SFC= 040000            US.PWF= 000010            U.RCTS= 000110            U2.R04= 100000            $HEADR= ****** GX
S4.VFL= 000010            US.RED= 000002            U.RED   000002            U2.SCS= 000004            $MBLUN= ****** GX
S5.ABP= 100000            US.SHR= 000001            U.RED2  000034            U2.SLV= 000200            $SAVRG= ****** GX
S5.BCC= 020000            US.SPU= 000002            U.SCB   000020            U2.VT5= 000002            $$$SYS= 004374
S5.DAO= 040000            US.UMD= 000010            U.SHST= 000076            U2.7CH= 010000            $$$SYX= 000000


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000514    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics
```

```
                                                  .SBTTL  GETMHC - GET AREA MINIMUM HOPS AND COST
                                          ;+
                                          ; ** - GETMHC - GET AREA MINIMUM HOPS AND COST
                                          ;
                                          ; INPUTS:
                                          ;       R1 = NODE-ID OF NEXT NODE TO AREA
                                          ;       R3 = MESSAGE BUFFER POINTER
                                          ;       R4 = CONTEXT AREA ADDRESS
                                          ;
                                          ; OUTPUTS:
                                          ;       R3 IS UPDATED
                                          ;       R0 IS CLOBBERED.
                                          ;-

000650                                    GETMHC: SAVRG   R1                      ; SAVE REGISTER
000652  005702                                    TST     R2                      ; IS THERE AN OUTPUT LINE ?
000654  001475                                    BEQ     40$                     ; IF EQ, NO
000656                                            SWSTK$  30$                     ;; ENTER SYSTEM STATE
000662  016402  000012                            MOV     A$XPT(R4),R2            ;; POINT TO XPT PDV
000666  016202  000016                            MOV     Z.DAT(R2),R2            ;; POINT TO PROCESS DDB
000672  001432                                    BEQ     10$                     ;; IN CASE SYSTEM GOING DOWN
                                          ;
                                          ; IF THIS IS A LEVEL 1 ROUTER, CHOOSE HOPS/COST TO NEAREST LEVEL 2 ROUTER
                                          ;
000674  032764  000100  000000                    BIT     #AF.LV2,A$FLG(R4)       ;; ARE WE A LEVEL 2 ROUTER?
000702  001004                                    BNE     5$                      ;; BR IF YES
000704  062702  000036                            ADD     #N$MHC1,R2              ;; POINT TO LEVEL 1 TABLES
000710  005001                                    CLR     R1                      ;; CORRESPONDING ENTRY TO LEVEL 2 ROUTER
000712  000404                                    BR      7$                      ;; JOIN COMMON CODE

000714  062702  000044                    5$:     ADD     #N$MHC2,R2              ;; POINT TO 3 WORD DESCRIPTOR
000720  016401  000004                            MOV     A$NUM(R4),R1            ;; PICK UP AREA NUMER

000724  020122                            7$:     CMP     R1,(R2)+                ;; ANY INFO FOR THIS AREA ?
000726  101014                                    BHI     10$                     ;; IF HI, NO INFORMATION
000730                                            MAP     (R2)+                   ;; ELSE MAP TO MHC VECTOR
000734  001411                                    BEQ     10$                     ;; IF EQ, NOT SET UP
000736  006301                                    ASL     R1                      ;; MAKE AREA NUMBER A WORD OFFSET
000740  061201                                    ADD     (R2),R1                 ;; CALCULATE ADDRESS OF ENTRY FOR AREA
000742                                            BIAS    R1                      ;; BIAS ADDRESS FOR APR 6
000752  011166  000000G                           MOV     (R1),R$R1(SP)           ;; PASS MHC INFO BACK IN SAVED R1
000756  000405                                    BR      20$                     ;; SUCCESS !

000760                                    10$:    RETC    R0                      ;; NO INFO - SET USER STATE CARRY
000772                                    20$:    RETURN                          ;; BACK TO USER STATE
000774  103425                            30$:    BCS     40$                     ; IF CS, NO INFORMATION

000776  012700  000000'                           MOV     #TEMP,R0                ; POINT TO BUFFER
001002  010146                                    MOV     R1,-(SP)                ; PUSH RETURNED DATA
001004  042701  176000                            BIC     #^C<1777>,R1            ; CLEAN OUT JUNK BITS
001010  010110                                    MOV     R1,(R0)                 ; LOAD UP BUFFER
001012                                            DBLOK$  PARAM,MP$DCO,DECUNS,2   ; STORE COST TO AREA
001022  012601                                    MOV     (SP)+,R1                ; PICK UP HOPS
001024  000301                                    SWAB    R1                      ; SHIFT INTO LOW BYTE
001026                                            ASR$    2,R1                    ; ...
001032  042701  177740                            BIC     #^C<37>,R1              ; ISOLATE HOPS
001036  010110                                    MOV     R1,(R0)                 ; LOAD UP BUFFER
```

```
                                                                 B 8
ARREOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:08       PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE                REFERENCES

Z.FLG     000010            #5-63        9-362
Z.LEN   = 000016            #5-63
Z.LLN     000006            #5-63
Z.MAP     000020            #5-63        9-343
Z.NAM     000004            #5-63        9-365
Z.PCB     000012            #5-63
Z.SCH     000007            #5-63
$CBTA   = ******    GX       9-375       9-382        9-391
$C5TA   = ******    GX       9-367
$HEADR  = ******    GX       6-143       8-295
$SAVRG  = ******    GX       6-92
$SAVVR  = ******    GX       9-330
.MBILD  = ******    GX       8-303       8-309        10-416
.MCODE  = ******    GX       10-419      10-421
```

```
                                                                 C 8
ARREOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:08       PAGE 5
MACRO CROSS REFERENCE                                     CREF    04.00
```

```
    189                                          .SBTTL  SAVROL - SAVE ROL NODE ADDRESSES
    190                                  ;+
    191                                  ; *** - SAVROL - SAVE ROL NODE ADDRESSES
    192                                  ;
    193                                  ; THIS ROUTINE IS CALLED AT SYSTEM STATE.
    194                                  ;
    195                                  ; INPUTS:
    196                                          R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
    197                                  ;
    198                                  ; OUTPUTS:
    199                                          R5 = UPDATED
    200                                  ;
    201                                  ;-
    202
    203 000250                           SAVROL: MAPXPT                          ;; MAP AND POINT TO XPT DATABASE
    204 000254  032764  000140  000000           BIT     #NF.LV1!NF.LV2,N$FLAG(R4) ;; IS IT AN ENDNODE ?
    205 000262  001440                           BEQ     30$                     ;; IF EQ, END NODE
    206 000264                                   MAP     N$ROA1+2(R2)            ;; MAP TO ROA1
    207 000272  016202  000026                   MOV     N$ROA1+4(R2),R2         ;; POINT TO ROL
    208 000276  001461                           BEQ     50$                     ;; IF EQ, ?
    209 000300                                   BIAS    R2                      ;; FORCE APR 6 MAPPING
    210 000310  005722                           TST     (R2)+                   ;; SKIP ZERO OFFSET ENTRY
    211 000312  005001                           CLR     R1                      ;; START WITH NODE 1
    212                                  ;
    213                                  ; SCAN ROL FOR REACHABLE NODES
    214                                  ;
    215 000314  005201                   10$:    INC     R1                      ;; NEXT NODE ADDRESS
    216 000316  017746  000000G                  MOV     @DECPT,-(SP)            ;; POINT TO DEC HOME BLOCK      ; RJK03
    217 000322  062716  000060                   ADD     #D$NN,(SP)              ;; POINT TO NUMBER OF NODES
    218 000326  020136                           CMP     R1,@(SP)+               ;; IS NODE ADDRESS IN TABLE ?
    219 000330  101044                           BHI     50$                     ;; IF HI, NO
    220 000332  005722                           TST     (R2)+                   ;; IS THIS NODE REACHABLE ?
    221 000334  001767                           BEQ     10$                     ;; IF EQ, NO
    222 000336  020164  000034                   CMP     R1,N$HROL(R4)           ;; IS ADDRESS HIGHER THAN ROL HIGHEST
    223 000342  101402                           BLOS    20$                     ;; IF LOS, NO - BRANCH
    224 000344  010164  000034                   MOV     R1,N$HROL(R4)           ;; ELSE SET HIGHEST ROL TO CURRENT ADDRESS
    225 000350  020164  000046           20$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
    226 000354  101757                           BLOS    10$                     ;; IF LOS, NO - GET NEXT ADDRESS
    227 000356                                   CALL    SAVCTX                  ;; ELSE SAVE THE ADDRESS IN CONTEXT AREA
    228 000362  000754                           BR      10$                     ;; LOOK FOR MORE
    229                                  ;
    230                                  ; END NODE - ADJACENT NODE INCLUDED FOR ACTIVE NODES
    231                                  ;
    232 000364  016202  000072           30$:    MOV     N$ADJ1(R2),R2           ;; PICK UP ADDRESS OF ADJACENCY
    233 000370  001424                           BEQ     50$                     ;; IF EQ, NOTHING UP
    234 000372                                   BIAS    R2                      ;; BIAS IT FOR APR 6
    235 000402  132762  000010  000002           BITB    #AT$UP,A$TYP(R2)        ;; IS IT UP ?
    236 000410  001414                           BEQ     50$                     ;; IF EQ, NOPE
    237 000412  016201  000000                   MOV     A$NID(R2),R1            ;; ELSE PICK UP NODE ADDRESS
    238 000416  020164  000034                   CMP     R1,N$HROL(R4)           ;; IS ADDRESS HIGHER THAN ROL HIGHEST
    239 000422  101402                           BLOS    40$                     ;; IF LOS, NO - BRANCH
    240 000424  010164  000034                   MOV     R1,N$HROL(R4)           ;; ELSE SET HIGHEST ROL TO CURRENT ADDRESS
    241 000430  020164  000046           40$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
    242 000434  101402                           BLOS    50$                     ;; IF LOS, NO - RETURN
    243 000436                                   CALL    SAVCTX                  ;; ELSE SAVE IT AND RETURN
    244 000442                           50$:    RETURN
```

```
Symbol table

ZS.BSY= 140000          Z.FLG   000010          Z.MAP   000020          Z.SCH   000007          $$SHFT= 000004
Z.AVL   000014          Z.LEN = 000016          Z.NAM   000004          $CEACC= ****** GX       $$$SYS= 004374
Z.DAT   000016          Z.LLN   000006          Z.PCB   000012          $SAVAL= ****** GX       $$$SYX= 000000
Z.DSP   000000

. ABS.  000270    000   (RW,I,GBL,ABS,OVR)
        001650    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 287
Work  file writes: 203
Size of work file: 38288 Words  ( 150 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:45.19
SY:BLNLST.V2,[135,134]BLNLST/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]BLNLST
```

```
                                        .SBTTL  BIT DEFINITIONS
                                ;
                                ; PARSE FLAG DEFINITIONS (L$PFG)
                                ;
        000001                          LP$UNT = 1                      ; UNIT NUMBER FOUND
        000002                          LP$TRB = 2                      ; TRIBUTARY NUMBER FOUND
        000004                          LP$MUX = 4                      ; DEVICE IS MUX
        000010                          LP$MPT = 10                     ; LINE IS MULTIPOINT
        000020                          LP$WDV = 20                     ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40                     ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100                    ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200                    ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK
                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                ;
        000001                          LF$REA = 1                      ; READ COUNTERS OPERATION
        000002                          LF$ZER = 2                      ; ZERO COUNTERS OPERATION
        000004                          LF$SKP = 4                      ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
        000010                          LF$VR2 = 10                     ; CONNECTED TO VERSION 2.0 NCP
        000020                          LF$END = 20                     ; EXECUTOR IS AN ENDNODE
        000040                          LF$SIG = 40                     ; SHOWING SIGNIFIGANT LINES/CIRCUITS
        040000                          LF$MLT = 40000                  ; MULTIPLE ADJACENCY FLAG
        100000                          LF$SEG  = 100000                ; SEGMENTED RESPONSE IN PROGRESS
                                ;
                                ; BIT POSITION DEFINITIONS
                                ;
        000001                          BIT0 = 1
        000002                          BIT1 = 2
        000004                          BIT2 = 4
        000010                          BIT3 = 10
        000020                          BIT4 = 20
        000040                          BIT5 = 40
        000100                          BIT6 = 100
        000200                          BIT7 = 200
        000400                          BIT8 = 400
        001000                          BIT9 = 1000
        002000                          BIT10 = 2000
        004000                          BIT11 = 4000
        010000                          BIT12 = 10000
        020000                          BIT13 = 20000
        040000                          BIT14 = 40000
        100000                          BIT15 = 100000
    64
```

```
                                        .SBTTL  CHKCIR - CHECK CIRCUIT-ID FORMAT
                                        ; OFFSETS INTO SCRATCH BUFFER (L$SCR) FOR DMC BASE TABLE COUNTERS
                                        ;
        000124                                  .ASECT
                000000                          .=0
        000000                          B$5:    .BLKB   1                       ; BASE TABLE + 5
        000001                          B$6:    .BLKB   1                       ; BASE TABLE + 6
        000002                          B$7:    .BLKB   1                       ; BASE TABLE + 7
        000003                          B$10:   .BLKB   1                       ; BASE TABLE + 10
        000004                          B$11:   .BLKB   1                       ; BASE TABLE + 11
        000005                          B$12:   .BLKB   1                       ; BASE TABLE + 12
        000006                          B$7S10: .BLKB   1                       ; SUM OF BASE TABLE 7 AND 10
                                                .EVEN
        000000                                  .PSECT
```

```
     53                                         .SBTTL  Macro calls and local data
     54                                 ;
     55                                 ; Macro library calls
     56                                 ;
     57                                         .MCALL  ASL$,CICCX$,IMSG$,MANDF$,PVCDF$,NTLDF$
     58                                         .MCALL  PDVDF$,RETC,RESRG,SAVRG,SLTDF$,UCBDF$
     59                                         .MCALL  HDRDF$,NSFDF$
     60
     61 000000                                  HDRDF$                  ; Define task header offsets
     62 000000                                  MANDF$                  ; Define network management symbols
     63 000000                                  NSFDF$                  ; Define NS: features mask
     64 000000                                  NTLDF$                  ; Define network loader symbols
     65 000000                                  PDVDF$                  ; Define PDV offsets
     66 000000                                  PVCDF$                  ; Define PVC descriptor
     67 000000                                  SLTDF$                  ; Define SLT offsets
     68 000000                                  UCBDF$                  ; Define UCB offsets
```

```
000000                          FIELD$
000000                           $BYTE   MU$PER,$EXIT,,MU$PER,+L$CUS          ; Permanent
000000                           $BYTE   MU$INC,$EXIT,,MU$INC,+L$CUS          ; Incoming
000000                           $BYTE   MU$OUT,$EXIT,,MU$OUT,+L$CUS          ; Outgoing
000000                           $NULL   ,,PVAERR
                        ;
                        ; DTE subexpression
                        ;
000000                          FIELD$  SDTE
000000                           $PARAM MP$DTE,,PAREND,MP$DTE,+L$PAR
000000                          FIELD$
000000                           $IMAGE MXDTE,$EXIT,DTEACT
000000                           $NULL   ,,PVAERR
                        ;
                        ; Maximum recalls subexpression
                        ;
000000                          FIELD$  SMXR
000000                           $PARAM MP$MXR,,PAREND,MP$MXR,+L$PAR
000000                          FIELD$
000000                           $MATCH 1,$EXIT,MXRACT
                        ;
                        ; Maximum window subexpression
                        ;
000000                          FIELD$  SMXW
000000                           $PARAM MP$MXW,,PAREND,MP$MXW,+L$PAR
000000                          FIELD$
000000                           $MATCH 1,$EXIT,MXWACT
                        ;
                        ; Number subexpression
                        ;
000000                          FIELD$  SNUM
000000                           $PARAM MP$NUM,,PAREND,MP$NUM,+L$PAR
000000                          FIELD$
000000                           $IMAGE MXDTE,$EXIT,NUMACT
000000                           $NULL   ,,PVAERR
                        ;
                        ; Recall timer subexpression
                        ;
000000                          FIELD$  SRET
000000                           $PARAM MP$RET,,PAREND,MP$RET,+L$PAR
000000                          FIELD$
000000                           $MATCH 2,$EXIT,RETACT
                        ;
                        ; CLEAR ALL SUB-EXPRESSION
                        ;
000000                          FIELD$  CALL
000000                           $EOM    ,CALL1

000000                          FIELD$  CALL1
000000                           $ERROR  ME.FUN
                        ;
                        ; CLEAR COUNTER TIMER SUB-EXPRESSION
                        ;
000000                          FIELD$  CLCT
000000                           $PARAM  MP$LCT,$EXIT
                        ;
                        ; Clear recall timer sub-expression
```

```
    837                                         .SBTTL  CHKDMP - CHECK FOR DMP DEVICE
    838
    839                                 ;+
    840                                 ;
    841                                 ; CHKDMP - CHECK FOR DMP DEVICE
    842                                 ;
    843                                 ; INPUTS:
    844                                 ;       .MSGPT - ADDRESS OF PORTION OF MESSAGE WHICH WAS MATCHED
    845                                 ;       .MSGCN - NUMBER OF BYTES MATCHED
    846                                 ;       R4     - ADDRESS OF CONTEXT AREA
    847                                 ;
    848                                 ; OUTPUTS:
    849                                 ;       IF DMP DEVICE:
    850                                 ;               R1 - VALUE OF PARAMETER
    851                                 ;       ELSE:
    852                                 ;               PARAMETER NOT APPLICABLE ERROR MESSAGE IS RETURNED
    853                                 ;
    854                                 ;-
    855
    856 000606  026427  000000  015436  CHKDMP: CMP     L$DDM(R4),#^RDMV        ; WAS DMV SPECIFIED?
    857 000614  001404                          BEQ     5$                      ; BR IF YES
    858 000616  026427  000000  015430          CMP     L$DDM(R4),#^RDMP        ; WAS DMP SPECIFIED?
    859 000624  001316                          BNE     PNAERR                  ; BR IF NO
    860 000626  016700  000000G         5$:     MOV     .MSGPT,R0               ; POINT TO MESSAGE
    861 000632  022767  000001  000000G         CMP     #1,.MSGCN               ; BYTE VALUE?
    862 000640  001002                          BNE     10$                     ; BR IF NO
    863 000642  111001                          MOVB    (R0),R1                 ; SAVE VALUE
    864 000644  000405                          BR      20$                     ; FINISH UP
    865 000646  005001                  10$:    CLR     R1                      ; STORE PARAMETER VALUE IN R1
    866 000650  156001  000001                  BISB    1(R0),R1                ; STORE HIGH BYTE
    867 000654  000301                          SWAB    R1                      ; MOVE INTO HIGH BYTE
    868 000656  151001                          BISB    (R0),R1                 ; STORE LOW BYTE
    869 000660                          20$:    RETURN
```

```
                                       .SBTTL  BSAACT - ACTIVE BASE ACTION ROUTINE

                               ;+
                               ;
                               ; BSAACT - ACTIVE BASE ACTION ROUTINE
                               ;
                               ; INPUTS:
                               ;       .MSGPT - POINTER TO 1 BYTE ACTIVE BASE
                               ;       R4     - POINTER TO CONTEXT AREA
                               ;
                               ; OUTPUTS:
                               ;       L$BSA(R4) - ACTIVE BASE VALUE
                               ;
                               ;-
001246                         BSAACT: CALL    CHKDMP                  ; CHECK FOR DMP
001252  110164  000122                 MOVB    R1,L$BSA(R4)            ; SAVE ACTIVE BASE VALUE
001256                                 RETURN
```

```
                                                        C  1
ALCHOP      CREATED BY  MACRO  ON 10-SEP-85 AT 11:46      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

$XLINK  = ******  GX      7-249      9-395
```

```
                                                        D  1
ALCHOP      CREATED BY  MACRO  ON 10-SEP-85 AT 11:46      PAGE 3
MACRO CROSS REFERENCE                                     CREF   04.00
```

```
                                                                        C 2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-2
Symbol table

MD.PER  000000      MF$ADD  000000      MO$PRS  000001      MP$CPU  000161      MP$LLE  000227
MD.RES  000004      MF$ADJ  177774      MO$REA  000200      MP$CSZ  001755      MP$LLO  004432
MD.ROB  000005      MF$ALL  177775      MO$SET  000000      MP$CUS  002127      MP$LMB  002152
MD.RSH  000013      MF$BYE  000300      MO$STA  000020      MP$CVA  000540      MP$LMX  002200
MD.SDU  000005      MF$CHA  000023      MO$SUM  000000      MP$DAL  002570      MP$LNA  000144
MD.SLO  000003      MF$DUM  000020      MO$TSK  000000      MP$DCO= 001464      MP$LNO  000233
MD.SNA  000000      MF$EVT  000001      MO$VOL  000000      MP$DDT  002177      MP$LOA  000170
MD.TLO  000004      MF$FUP  000264      MO$ZER  000000      MP$DEL  001131      MP$LOG  004514
MD.UNA  000002      MF$KNO  177777      MP$ACB  000012      MP$DES  000156      MP$LOO  000620
MD.UNR  000003      MF$LOA  000017      MP$ACC  000514      MP$DEV  002114      MP$LTM  001613
MD.UOB  000007      MF$LOO  177775      MP$ACT  004526      MP$DFA  001320      MP$LTY  004533
MD.VOL  000006      MF$REA  000024      MP$ADD  000766      MP$DHO= 001465      MP$LWI  000230
ME$ALI  000010      MF$SIG  177773      MP$ADJ  001440      MP$DIA  000173      MP$MAC  004420
ME$AL2  000003      MF$SPF  000302      MP$ADP  006325      MP$DLB  002571      MP$MAD  001630
ME$ARE  000005      MF$SYS  000026      MP$ADS  004406      MP$DLG  004521      MP$MAP  006323
ME$CIR  000003      MF$TES  000022      MP$ALB  000036      MP$DLI= 001466      MP$MAR  001635
ME$EXA  000000      MF$TRA  000301      MP$AMC  001640      MP$DLT  002200      MP$MAV  006321
ME$EXE  000200      MF$TRI  000021      MP$AMH  001641      MP$DRO  001441      MP$MBN  001636
ME$LIN  000001      MF$ZER  000025      MP$ANB  004420      MP$DST  000454      MP$MBR  001637
ME$LOG  000002      ML$ALL  100000      MP$ASB  000024      MP$DTE  002140      MP$MBU  001642
ME$MOD  000004      ML$CLS  000000      MP$ASC  004432      MP$DTY= 001452      MP$MCB  000156
ME$NOD  000000      ML$CON  000001      MP$AUS  000512      MP$DUA  000207      MP$MCO  001632
ME$NON  177777      ML$EXT  000001      MP$BBT  002165      MP$DUC  000210      MP$MDE  004421
ME$OBJ  000007      ML$FIL  000002      MP$BDF  002164      MP$DUM  000202      MP$MHO  001633
ME$OB2  000004      ML$FIR  000000      MP$BFQ  002121      MP$DUP  002127      MP$MLB  000202
ME$PRO  000005      ML$INT  000000      MP$BLK  001616      MP$DVC  002114      MP$MLK  001306
ME$SYS  000006      ML$KNO  140000      MP$BLO  001452      MP$DWE  001321      MP$MLN  001631
ME.BLO  177744      ML$MON  000003      MP$BMX  002176      MP$ELT  000157      MP$MLP  006333
ME.CON  177753      ML$SYS  000002      MP$BNP  006330      MP$ETY  001605      MP$MRB  002171
ME.CST  177765      ML$TOP  000001      MP$BRT  001620      MP$EVE  000311      MP$MRO  001605
ME.DIS  177755      MN$UNL  000377      MP$BSA  002176      MP$FNC  001752      MP$MRP  006322
ME.DON  177600      MO$ACC  000200      MP$BSD  002203      MP$GDT  002222      MP$MRT  002153
ME.FCO  177762      MO$ADD  000002      MP$BSI  002200      MP$GNM  002223      MP$MRV  006320
ME.FIO  177756      MO$ALA  000004      MP$BSP  006331      MP$GRO  002115      MP$MSB  000170
ME.FOP  177763      MO$ALI  000000      MP$BUF  000170      MP$GRP  000541      MP$MVE  000145
ME.FOR  177776      MO$CHA  000040      MP$BUP  006324      MP$GTY  002224      MP$MVI  001634
ME.FUN  177777      MO$CIR  000003      MP$BUS  001643      MP$HAD  001757      MP$MVR  001751
ME.GRO  177745      MO$CLE  000100      MP$CAC  001750      MP$HBT  002142      MP$MWN  002154
ME.HAR  177750      MO$COU  000060      MP$CAS  001762      MP$HDD  000162      MP$MXB  002172
ME.IID  177767      MO$DAC  000000      MP$CAT  002210      MP$HTM  001612      MP$MXC  000466
ME.LCO  177766      MO$DEF  000000      MP$CCS  004406      MP$HWA  002210      MP$MXR  001630
ME.LPR  177757      MO$DPR  000000      MP$CHN  002141      MP$IAT  001322      MP$MXW  002143
ME.MPR  177773      MO$ENT  000017      MP$CIR  000144      MP$IDE  000144      MP$NAA  006335
ME.MVE  177771      MO$EVE  000100      MP$CLK  002131      MP$IDP  006327      MP$NAC  001130
ME.OPE  177747      MO$INF  000160      MP$CLN  002126      MP$IHO  000215      MP$NAP  006326
ME.PLO  177751      MO$INS  000002      MP$CLT  002211      MP$INA  002177      MP$NCT  000240
ME.PMI  177743      MO$LIN  000001      MP$CMB  002142      MP$IND  002204      MP$NET  002114
ME.PNA  177752      MO$LOG  000001      MP$CMK  000537      MP$INI  002201      MP$NLI  000765
ME.PRI  177775      MO$MIR  000031      MP$CMX  002153      MP$ITI  000776      MP$NNA  000764
ME.PTY  177772      MO$NAM  000001      MP$CND  000310      MP$LAA  000231      MP$NOD  000500
ME.PVA  177760      MO$NIC  000023      MP$CNU  001753      MP$LAN  000234      MP$NRA  006334
ME.RES  177761      MO$NOD  000000      MP$COB  000311      MP$LAR  000222      MP$NRB  000214
ME.ROO  177754      MO$OFF  000001      MP$CON  002126      MP$LAS  000012      MP$NSA  001617
ME.SIZ  177774      MO$ON   000000      MP$COS  001604      MP$LBS  000203      MP$NTI  002141
ME.SYS  177746      MO$OPT  000001      MP$CPF  000230      MP$LCO  000226      MP$NUM  001642
ME.UCO  177770      MO$PER  000200      MP$CPL  000232      MP$LCT  000156      MP$NVE  001274
MF$ACT  177776      MO$PRO  000002      MP$CPT  000226      MP$LHL  000232      MP$NXN= 001476
```

```
                                                                        D 2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-3
Symbol table

MP$OAC  000632      MP$SEP  000144      MS$ALO  000007      MT$MAX  000037      M$$CRB= 000124
```

```
                                                .SBTTL  FMTSCO - FORMAT SCOPE PARAMETER
                                        ;+
                                        ; **-FMTSCO-FORMAT SCOPE PARAMETER
                                        ;
                                        ; THIS ROUTINE IS CALLED TO FORMAT THE SCOPE PARAMETER FIELD.
                                        ;
                                        ; INPUTS:
                                        ;       R1 = POINTS TO SCOPE (0 = GLOBAL, ELSE UCB ADDRESS OF TERMINAL)
                                        ;       R3 = BUFFER TO HOLD FORMATED FIELD
                                        ;
                                        ; OUTPUTS:
                                        ;       R3 = NEXT FREE BYTE IN BUFFER
                                        ;
                                        ; REGISTERS:
                                        ;       R4 IS PRESERVED
                                        ;-

 000354                                 FMTSCO:
 000354  112723  000144                         MOVB    #MP$SCO,(R3)+           ; SET PARAMETER TYPE = SCOPE
 000360  105023                                 CLRB    (R3)+                   ; ...
 000362  010300                                 MOV     R3,R0                   ; SAVE POINTER TO BUFFER
 000364  011103                                 MOV     (R1),R3                 ; GET SCOPE
 000366  001006                                 BNE     10$                     ; IF NE, TERMINAL SCOPE
                                        ;
                                        ; FORMAT GLOBAL SCOPE
                                        ;
 000370  112720  000301                         MOVB    #301,(R0)+              ; DATA TYPE =  CODED, MULT FIELD (1)
 000374  112720  000201                         MOVB    #201,(R0)+              ; SCOPE TYPE DATA TYPE
 000400  105020                                 CLRB    (R0)+                   ; SCOPE TYPE = GLOBAL
 000402  000417                                 BR      20$                     ; RETURN
                                        ;
                                        ; FORMAT TERMINAL SCOPE
                                        ;
 000404  112720  000302                 10$:    MOVB    #302,(R0)+              ; DATA TYPE = CODED, MULT FIELD (2)
 000410  112720  000201                         MOVB    #201,(R0)+              ; SCOPE TYPE DATA TYPE
 000414  112720  000001                         MOVB    #1,(R0)+                ; SCOPE TYPE = TERMINAL
 000420  112720  000100                         MOVB    #100,(R0)+              ; TERMINAL ID DATA TYPE = ASCII IMAGE
 000424  105020                                 CLRB    (R0)+                   ; ALLOCATE IMAGE FIELD LENGTH BYTE
 000426  010005                                 MOV     R0,R5                   ; SAVE POINTER TO START OF STRING
 000430                                         CALL    $FMTDV                  ; FORMAT TERMINAL DEVICE NAME STRING
 000434  010003                                 MOV     R0,R3                   ; POINT TO END OF STRING
 000436  160503                                 SUB     R5,R3                   ; CALC STRING LENGTH
 000440  110345                                 MOVB    R3,-(R5)                ; SET LENGTH OF IMAGE STRING

 000442  010003                         20$:    MOV     R0,R3                   ; POINT TO NEXT FREE BYTE IN BUFFER
 000444                                         RETURN
```

```
                                                SBTTL  FMTDES - FORMAT DESTINATION PARAMETER
```

```
     53                                                   .SBTIL  MACRO CALLS AND LOCAL DATA
     54                                           ;
     55                                           ; MACRO LIBRARY CALLS
     56                                           ;
     57                                                   .MCALL  ANBDF$,MANDF$,ALUN$S,SWSTK$,RETC,SAVMAP,RESMAP,MAP
     58                                                   .MCALL  HDRDF$,UCBDF$,ALCCX$,SAVRG,RESRG
     59
     60 000000                                            ANBDF$                          ; DEFINE ALIAS NAME BLOCK OFFSETS
     61 000000                                            MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     62 000000                                            UCBDF$                          ; DEFINE UCB OFFSETS
     63 000000                                            HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     64
     65 000000                                            ALCCX$  LIST                    ; DEFINE CONTEXT AREA
                                                  ;
                                                  ; CONTEXT AREA OFFSET DEFINITIONS
                                                  ;
                000020                                    MAXID=16.                       ; MAXIMUM LENGTH OF USER ID
                000010                                    MAXPAS=8.                       ; MAXIMUM LENGTH OF PASSWORD
                000020                                    MAXACC=16.                      ; MAXIMUM LENGTH OF ACCOUNT
        000020                                            .ASECT
                000000                                    .=0
                                                  ;
                                                  ; OFFSETS IN COMMON WITH READ INFORMATION MODULES
                                                  ;
        000000                                    A$FLG:  .BLKW   1                       ; FLAG WORD
        000002                                    A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
        000004                                    A$PAR:  .BLKW   1                       ; CURRENT PARAMETER TYPE
        000006                                    A$BUF:  .BLKW   1                       ; SAVED POINTER TO BUFFER
        000010                                    A$NUM:  .BLKB   1                       ; NUMBER OF BYTES IN ANB (FOR SET)
        000011                                    A$TYP:  .BLKB   1                       ; ALIAS FORMAT TYPE
        000012                                    A$ALI:  .BLKW   3                       ; ALIAS NAME, SPACE FILLED
        000020                                    A$SCO:  .BLKW   1                       ; SCOPE OF ALIAS
                                                  ;
                                                  ; OFFSETS UNIQUE TO CHANGE PARAMETERS MODULES
                                                  ;
        000022                                    A$DES:  .BLKW   3                       ; DESTINATION NAME, SPACE FILLED
        000030                                            .BLKB   <1+MAXID>+<1+MAXPAS>+<1+MAXACC> ; ACCESS CONTROL STORAGE
        000103                                    A$LEN:                                  ; LENGTH OF CONTEXT AREA
        000000                                            .PSECT
                                                  ;
                                                  ; FLAGS WORD BIT DEFINITIONS
                                                  ;
                000001                                    AF.INI=1                        ; SUCCESSFUL INITIALIZATION COMPLETED
                000002                                    AF.OVR=2                        ; CONTEXT AREA OVERFLOWED
     66
     67                                           ;
     68                                           ;       LOCAL DATA
     69                                           ;
     70                                           ; Table of Aliases
     71                                           ;
     72                                                   .NLIST BEX
     73
     74 000000         006                        ALIT1:  .BYTE   6
     75 000001         040     040     040                .ASCII  /      /
     76                                                   .EVEN
     77 000010         005                        ALIT2:  .BYTE   5
     78 000011         044     110     117                .ASCII  /$HOST/
```

```
                                                                   C  5
ALXXIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE            REFERENCES

U.CW1     000010          7-314
U.RED     000002          7-305       7-307
$COTB   = ******   GX     7-278
$HEADR  = ******   GX     7-316
$TMLUN  = ******   GX     7-285       7-301
```

```
                                                                   D  5
ALXXIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 3
MACRO CROSS REFERENCE                                   CREF    04.00
```

                                         C 6
```
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 6-7
Symbol table


Work  file  reads: 222
Work  file writes: 160
Size of work file: 32149 Words  ( 126 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:29.10
SY:ARREIN.V2,[135,134]ARREIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ARREIN
```

                                         D 6
```
ARREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
     001040                        DBLOK$  PARAM,MP$DHO,DECUNS,1   ; STORE HOPS TO NODE

     001050              40$:      RESRG   R1                      ; RESTORE REGISTER
     001052                        RETURN

```

```
                                   SBTTL   FMTCIR - FORMAT CIRCUIT ID
```

```
                                                                 C  8
ARREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:08      PAGE 5

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME          REFERENCES

ADJDF$              #5-56          5-59
ARRCX$              #5-56          5-67
ASR$                #5-57          8-306
BIAS                #5-57          6-129          6-136          7-216          7-231          7-234          8-291
CALL                 6-109         6-118          6-171          6-172          6-173          8-269          9-367          9-375          9-382          9-391
                     10-420        11-455
CNTR$$              #5-62
DATA$$              #5-62          8-303          8-309          10-416         10-419         10-421
DBLOK$              #5-62          8-303          8-309          10-416         10-419         10-421
DHBDF$              #5-56          5-60
MANDF$              #5-56          5-61          #5-62           5-62
MAP                 #5-57          6-126          6-135          7-212          7-228          7-235          8-287
MBILD$              #5-56          5-62
PDVDF$              #5-56          5-63
RESMAP              #5-57          7-249
RESRG               #5-57          7-248          8-311          11-464
RETC                #5-57          6-143          8-295
RETURN               6-144         6-186          7-250          8-296          8-312          9-396          10-423         11-465
SAVMAP              #5-57          7-201
SAVRG               #5-57          7-202          8-266          11-440
SLTDF$              #5-56          5-64
SWSTK$               6-109         8-269
XPDDB$              #5-56          5-65
```

```
                                                                 D  8
**FILE**ID**BLNLST
```

```
                                                .SBTTL  SAVREM - SAVE REMOTE NODE ADDRESSES
                                        ;+
                                        ; *** - SAVREM - SAVE REMOTE NODE ADDRESSES
                                        ;
                                        ; THIS ROUTINE IS CALLED AT SYSTEM STATE AND MUST HAVE PREVIOUSLY SAVED KISAR6.
                                        ;
                                        ; INPUTS:
                                        ;       R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R5 = UPDATED
                                        ;
                                        ;-

000444                                  SAVREM:
000444  017702  000000G                         MOV     @DECPT,R2                                       ; RJK03
000450  062702  000002                          ADD     #D$RNN,R2               ;; POINT TO START OF REMOTE LIST
                                        ;
                                        ; SCAN REMOTE TABLES
                                        ;
000454  011246                          10$:    MOV     (R2),-(SP)              ;; SET UP TO MAP TO RNB
000456                                          CALL    @CEACC                  ;; MAP TO RNB                   ; RJK03
000462  012602                                  MOV     (SP)+,R2                ;; GET MAPPED ADDRESS
000464  001425                                  BEQ     30$                     ;; IF EQ, NONE

000466  016201  000010                  15$:    MOV     R.ADD(R2),R1            ;; GET NODE ADDRESS
000472  032762  100000  000012                  BIT     #RF.LOO,R.FLAG(R2)      ;; LOOP NODE ?
000500  001365                                  BNE     10$                     ;; IF NE, YES - DON'T SAVE IT
000502  020164  000032                          CMP     R1,N$HREM(R4)           ;; IS ADDRESS HIGHER THAN REM HIGHEST
000506  101402                                  BLOS    20$                     ;; IF LOS, NO - BRANCH
000510  010164  000032                          MOV     R1,N$HREM(R4)           ;; ELSE SET HIGHEST REM TO CURRENT ADDRESS
000514  020164  000046                  20$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
000520  101755                                  BLOS    10$                     ;; IF LOS, NO - GET NEXT ADDRESS
000522                                          CALL    SAVCTX                  ;; SAVE THE NODE ADDRESS
000526  103352                                  BCC     10$                     ;; IF CC, MORE ROOM
000530  032764  040000  000000                  BIT     #NF.FTT,N$FLAG(R4)      ;; IS THIS FIRST TIME THROUGH LIST
000536  001346                                  BNE     10$                     ;; IF NE, YES - SEARCH WHOLE LIST TO
                                                                                ;;      GET HIGHEST REMOTE NODE ADDRESS
000540                                  30$:    RETURN
```

```
                                                .SBTTL  SAVECL - SAVE ECL NODES
```

```
                                                                  C 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09       PAGE 1
SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL  VALUE              REFERENCES

AT$UP   = 000010            10-235      14-400      14-417
A$LEN     000004            14-409
A$NID     000000            10-237      14-402      14-419
A$TYP     000002            10-235      14-400      14-417
BIT0    = 000001           #6-87
BIT1    = 000002           #6-87
BIT10   = 002000           #6-87
BIT11   = 004000           #6-87
BIT12   = 010000           #6-87
BIT13   = 020000           #6-87
BIT14   = 040000           #6-87
BIT15   = 100000           #6-87
BIT2    = 000004           #6-87
BIT3    = 000010           #6-87
BIT4    = 000020           #6-87
BIT5    = 000040           #6-87
BIT6    = 000100           #6-87
BIT7    = 000200           #6-87
BIT8    = 000400           #6-87
BIT9    = 001000           #6-87
BLNLST    000054 RG         8-132     #9-153
B$10      000003           #5-87
B$11      000004           #5-87
B$12      000005           #5-87
B$5       000000           #5-87
B$6       000001           #5-87
B$7       000002           #5-87
B$7S10    000006           #5-87
CEACC   = ******   GX       11-267      13-340
DECPT   = ******   GX       10-216      11-261      13-334      14-392      15-441
DISTBL    000000 R         #7-104       9-156
D$NBEA    000056            14-395
D$NBRA    000054            14-394
D$NLN     000030            14-393
D$NN      000060            10-217
D$RNN     000002            11-262      13-335
D$SER     000032            15-443
E$NBR     000014           #5-68
E$NBS     000020           #5-68
E$NCR     000034           #5-68
E$NCS     000036           #5-68
E$NIC     000044           #5-68
E$NLEN    000050           #5-68
E$NLLA    000012           #5-68
E$NLNK    000000           #5-68
E$NML     000040           #5-68
E$NMR     000024           #5-68
E$NMS     000030           #5-68
E$NNOD    000002           #5-68        12-305      12-307
E$NRT     000042           #5-68
E$NR1P    000005           #5-68
E$NSEG    000010           #5-68
```

```
                                                                  D 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09       PAGE 2
SYMBOL CROSS REFERENCE                                      CREF   04.00
```

```
     66                                                  .SBTTL  CHKCIR - CHECK CIRCUIT-ID FORMAT
     67                                          ;+
     68                                          ; **-CHKCIR-CHECK CIRCUIT-ID FORMAT
     69                                          ;
     70                                          ; THIS ROUTINE CHECKS THE PASSED CIRCUIT-ID FORMAT FOR VALIDITY,
     71                                          ; AND IF VALID, SAVES THE CIRCUIT-ID CONTEXT.
     72                                          ;
     73                                          ; INPUTS:
     74                                          ;        R1 = CIRCUIT-ID ADDRESS
     75                                          ;        R4 = ADDRESS OF CONTEXT AREA
     76                                          ;
     77                                          ; OUTPUTS.
     78                                          ;        C-BIT ERROR INDICATOR
     79                                          ;          IF CLEAR, L$TYP(R4) = SAVED CIRCUIT-ID TYPE AND:
     80                                          ;             IF TYPE >0 , CIRCUIT-ID PARSED
     81                                          ;          IF SET, ERROR R0 CONTAINS ERROR CODE
     82                                          ;
     83                                          ; REGISTERS:
     84                                          ;        ALL REGISTERS ARE PRESERVED UNLESS THERE IS AN ERROR (SEE ABOVE)
     85                                          ;-
     86
     87 000000                                   CHKCIR::
     88                                          ;
     89                                          ; CHECK CIRCUIT FORMAT TYPE
     90                                          ;
     91 000000  121127  177773                           CMPB    (R1),#MF$SIG            ; SIGNIFIGANT CIRCUITS ?
     92 000004  001403                                   BEQ     5$                      ; IF EQ, YES
     93 000006  121127  177776                           CMPB    (R1),#MF$ACT            ; IS IT WITHIN VALID RANGE ?
     94 000012  002414                                   BLT     20$                     ; IF LT, NO - ERROR
     95 000014  152764  000363  000041   5$:             BISB    #LP$WDV!LP$WCN!LP$WUN!LP$WTR!LP$UNT!LP$TRB,L$PFG(R4)
     96                                                                                  ; ASSUME ACT OR KNOWN CIRCUITS
     97                                                                                  ;(WILD EVERYTHING)
     98 000022  111164  000063                           MOVB    (R1),L$TYP(R4)          ; SAVE CIRCUIT FORMAT TYPE
     99 000026  002404                                   BLT     10$                     ; IF LT, NO CIRCUIT NAME - RETURN
    100 000030  001405                                   BEQ     20$                     ; IF EQ, INVALID ID - ERROR
    101                                          ;
    102                                          ; PARSE THE CIRCUIT-ID STRING
    103                                          ;
    104 000032                                           CALL    PRSCIR                  ; PARSE THE CIRCUIT-ID
    105 000036  000405                                   BR      30$                     ; AND RETURN
    106 000040  000241                           10$:    CLC                             ; SET SUCCESS
    107 000042  000403                                   BR      30$                     ; AND RETURN
    108 000044  012700  177767                   20$:    MOV     #ME.IID,R0              ; INVALID CIRCUIT-ID
    109 000050  000261                                   SEC                             ; ...
    110 000052                                   30$:    RETURN
    111
    112         000001                                   .END
```

```
                                                .SBTTL  BIT DEFINITIONS
                                        ;
                                        ; PARSE FLAG DEFFINITIONS (L$PFG)
                                        ;
              000001                            LP$UNT = 1                          ; UNIT NUMBER FOUND
              000002                            LP$TRB = 2                          ; TRIBUTARY NUMBER FOUND
              000004                            LP$MUX = 4                          ; DEVICE IS MUX
              000010                            LP$MPT = 10                         ; LINE IS MULTIPOINT
              000020                            LP$WDV = 20                         ; WILD CARD DEVICE NAME FOUND
              000040                            LP$WCN = 40                         ; WILD CARD CONTROLLER NUMBER FOUND
              000100                            LP$WUN = 100                        ; WILD CARD UNIT NUMBER FOUND
              000200                            LP$WTR = 200                        ; WILD CARD TRIBUTARY NUMBER FOUND
              000360                            LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK
                                        ;
                                        ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                        ;
              000001                            LF$REA = 1                          ; READ COUNTERS OPERATION
              000002                            LF$ZER = 2                          ; ZERO COUNTERS OPERATION
              000004                            LF$SKP = 4                          ; SKIP NEXT ''FIND NEXT LINE'' OPERATION.
                                                                                    ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                                    ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                                    ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                                    ; OF THE TRIBUTARY COUNTERS.
              000010                            LF$VR? = 10                         ; CONNECTED TO VERSION 2.0 NCP
              000020                            LF$END = 20                         ; EXECUTOR IS AN ENDNODE
              000040                            LF$SIG = 40                         ; SHOWING SIGNIFIGANT LINES/CIRCUITS
              040000                            LF$MLT = 40000                      ; MULTIPLE ADJACENCY FLAG
              100000                            LF$SEG  = 100000                    ; SEGMENTED RESPONSE IN PROGRESS
                                        ;
                                        ; BIT POSITION DEFINITIONS
                                        ;
              000001                            BIT0 = 1
              000002                            BIT1 = 2
              000004                            BIT2 = 4
              000010                            BIT3 = 10
              000020                            BIT4 = 20
              000040                            BIT5 = 40
              000100                            BIT6 = 100
              000200                            BIT7 = 200
              000400                            BIT8 = 400
              001000                            BIT9 = 1000
              002000                            BIT10 = 2000
              004000                            BIT11 = 4000
              010000                            BIT12 = 10000
              020000                            BIT13 = 20000
              040000                            BIT14 = 40000
              100000                            BIT15 = 100000
     64
```

```
     70                                                 .SBTTL  Context area
     71
     72 000000                                          CICCX$  LIST
                                                ;
                                                ; CONTEXT AREA OFFSET DEFINITIONS
                                                ;
        000004                                          .ASECT
                        000000                          . = 0
                        000020                          L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                                ;
                                                ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                                ; as in the CIRCX$ and LOCCX$ macros.
                                                ;
        000000                                  L$NAM:
        000000                                  L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                                  L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                                  L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                                  L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                        000020                          . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                                  L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                                  L$TRB:  .BLKB   1               ; TRIBUTARY NUMBER
        000041                                  L$PFG:  .BLKB   1               ; PARSE FLAGS
        000042                                  L$SLT:  .BLKW   1               ; Current SLT/PVC address
        000044                                  L$NXT:  .BLKW   1               ; Pointer to next SLT/PVC address
        000046                                  L$SNM:  .BLKW   1               ; REMAINING NUMBER OF SYSTEM LINES
        000050                                  L$TPT:  .BLKW   1               ; Current tributary pointer (-1 for PSI)
        000052                                  L$TNM:  .BLKB   1               ; REMAINING NUMBER OF TRIBUTARIES
        000053                                  L$CTB:  .BLKB   1               ; CURRENT TRIBUTARY NUMBER
        000054                                  L$CHN:  .BLKB   1               ; Channel number / X.25 port number and
        000055                                  L$PDV:  .BLKB   1               ; ... PDV assigned
        000056                                  L$MSG:  .BLKW   1               ; ERROR MESSAGE STRING POINTER
        000060                                  L$BUF:  .BLKW   1               ; SAVED BUFFER POINTER
        000062                                  L$OPT:  .BLKB   1               ; SAVED OPTIONS BYTE
        000063                                  L$TYP:  .BLKB   1               ; CIRCUIT-ID FORMAT TYPE
        000064                                  L$FLG:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000066                                  L$FL1:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000070                                  L$FL2:  .BLKW   1               ; FLAG WORD (FOR CURRENT CIRCUIT)
        000072                                  L$PAR:  .BLKW   1               ; CURRENT PARAMETER TYPE
        000074                                  L$SLEN: .BLKB   1               ; Significant length of circuit name
        000075                                  L$PRO:                          ; Line protocol
        000075                                  L$MTYP: .BLKB   1               ; Network management circuit type
        000076                                  L$FLX:  .BLKW   1               ; Flags word for X.25 circuit commands
        000100                                  L$STA:  .BLKB   1               ; CIRCUIT STATE
        000101                                  L$COS:  .BLKB   1               ; CIRCUIT COST
        000102                                  L$OWN:  .BLKB   1               ; CIRCUIT OWNER (PDV INDEX)
        000103                                  L$TAD:  .BLKB   1               ; TRIBUTARY ADDRESS
        000104                                  L$ACT:  .BLKB   1               ; MULTIPOINT ACTIVE RATIO
        000105                                  L$DEA:  .BLKB   1               ; MULTIPOINT DEAD RATIO
        000106                                  L$DDT:  .BLKW   1               ; DMP DEAD TIMER
        000110                                  L$DLT:  .BLKW   1               ; DMP DELAY TIMER
        000112                                  L$PLT:  .BLKW   1               ; DMP PO'L TIMER
        000114                                  L$BBT:  .BLKW   1               ; DMP BABBLE TIMER
        000116                                  L$NMT:  .BLKW   1               ; DMP NOPMAL TIMER
        000120                                  L$XMT:  .BLKW   1               ; DMP TRANSMIT TIMER
        000122                                  L$BSA:  .BLKB   1               ; DMP ACTIVE BASE
        000123                                  L$BSD:  .BLKB   1               ; DMP DYING BASE
        000124                                  L$BSI:  .BLKB   1               ; DMP INACTIVE BASE
```

```
    495                                 ;
    496 000000                                  FIELD$  CRET
    497 000000                                   $PARAM  MP$RET,$EXIT
    498
    499                                   .ENDC ; .IF NDF R$$11S
    500
    501 000000                                  FIELD$
```

```
    871                                                          .SBTTL  CHNACT - X.25 PVC Channel Number action routine
    872                                                  ;+
    873                                                  ;  ** CHNACT - X.25 PVC Channel Number action routine
    874                                                  ;
    875                                                  ; INPUTS
    876                                                  ;        .MSGPT   - Pointer to channel number
    877                                                  ;        R4       - Pointer to context area
    878                                                  ;
    879                                                  ; OUTPUTS
    880                                                  ;        L$XCH(R4) Contains channel number
    881                                                  ;        Error taken if channel not in range 0-4095
    882                                                  ;-
    883
    884 000662                                           CHNACT:
    885 000662  016701  000000G                                  MOV     .MSGPT,R1               ; Get pointer to channel number
    886 000666  112164  000232                                   MOVB    (R1)+,L$XCH(R4)         ; Get channel number
    887 000672  111164  000233                                   MOVB    (R1),L$XCH+1(R4)
    888 000676  026427  000232  007777                           CMP     L$XCH(R4),#MAXCHN       ; Check if channel number in range
    889 000704  101271                                           BHI     PVAERR                  ; Take error exit if out of range
    890 000706                                                   RETURN
```

```
                                        .SBTTL  BSDACT - DYING BASE ACTION ROUTINE

                                ;+
                                ;
                                ; BSDACT - DYING BASE ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 1 BYTE DYING BASE
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$BSD(R4) - DYING BASE VALUE
                                ;
                                ;-
001260                          BSDACT: CALL    CHKDMP                  ; CHECK FOR DMP
001264  110164  000123                  MOVB    R1,L$BSD(R4)            ; SAVE DYING BASE VALUE
001270                                  RETURN
```

```
                                                    D  1
ALCHOP      CREATED BY  MACRO  ON 10-SEP-85 AT 11:46     PAGE 3

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME          REFERENCES

ALCCX$              #5-58      5-70
ANBDF$              #5-57      5-60
CALL                 6-105     6-107      7-158      7-166      7-170      7-195      7-214      7-238      8-303      8-313
                     8-322     8-331      8-338      9-396      9-399      10-433
CALL).              #5-58      7-249      9-395
DHBDF$              #5-58      5-63
HDRDF$              #5-58      5-62
MANDF$              #5-57      5-61
MAP                 #5-57
RESMAP              #5-57      7-256      8-341
RESRG               #5-57      9-401
RETURN               6-117     7-265      8-350      9-402      10-449
SAVMAP              #5-57      7-159      8-304
SAVRG               #5-57      9-373
SWSTK$              #5-57      7-158      8-303
```

```
                                                    E  1
**FILE**ID**ALREIN
```

```
                                                                           D 2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-7
Symbol table

MP$OAC  000632     MP$SER  000144     MS$ALO  000007     MT$MAX  000037     M$$CRB= 000124
MP$OAN  000620     MP$SE1  000000     MS$ASE  000006     MT$MUL  000100     M$$CRX= 000000
MP$OCO  000776     MP$SGZ  001644     MS$ATR  000011     MT$NLE  000017     M$$FCS= 000000
MP$OHO  000214     MP$SID  000176     MS$AUT  000000     MT$NON  000001     M$$MGE= 000000
MP$ONA  000764     MP$SIN  000310     MS$CLE  000003     MT$NR4  000005     M$$NET= 000000
MP$ONR  002114     MP$SLI  000156     MS$DED  000004     MT$NTY  000060     M$$OVR= 000000
MP$OQL  001453     MP$SLO  000171     MS$DIE  000003     MT$OCT  000060     NM.CLN  003400
MP$OTI  000777     MP$SMX  002202     MS$DUM  000004     MT$PHA  000002     NM.INI  002400
MP$OTY  001022     MP$SND  000500     MS$FAI  000013     MT$POI  000000     NM.OPR  003000
MP$OUS  000777     MP$SNP  006332     MS$GLO  000000     MT$QP2  000010     NM.VR2  002401
MP$OVE  001010     MP$SNU  000542     MS$HOL  000002     MT$ROU  000000     NM.VR3  002402
MP$OWN  004374     MP$SOB  000524     MS$INA  000002     MT$R04  000004     N$$ACC= 000001
MP$PAR  000036     MP$SPA  000157     MS$LOA  000003     MT$SEC  000000     N$$BUF= 000001
MP$PAS  000513     MP$SPR  000536     MS$LOO  000002     MT$SGD  000020     N$$LDV= 000001
MP$PCH  002152     MP$SPS  000513     MS$OFF  000001     MT$SYS  000002     N$$MCP= 000001
MP$PCO  000024     MP$STA  000000     MS$ON   000000     MT$TER  000001     N$$MLL= 000001
MP$PCT  000144     MP$STI  002140     MS$PRO  000006     MT$TRI  000002     N$$MOV= 000010
MP$PDT  002114     MP$STT  002201     MS$REA  000004     MT$TYP  007777     N$$NCT= 000001
MP$PHA  000170     MP$STY  000175     MS$REF  000001     MT$USD  000000     N$$PEM= 000001
MP$PHY  000012     MP$SUB  000001     MS$RES  000003     MT$X25  000003     P$$P45= 000000
MP$PLI  000025     MP$SUR  000156     MS$RST  000001     MU$INC  000001     P$$WRD= 000000
MP$PLN  002140     MP$SUS  000512     MS$RSX  000002     MU$OUT  000002     Q$$OPT= 000010
MP$PLD  000012     MP$SVR  000163     MS$RT   000005     MU$PER  000000     R$$DER= 000000
MP$PMC  004374     MP$SWI  002260     MS$SER  000002     MV$III  000000     R$$K11= 000001
MP$PNT  002126     MP$SYP  002424     MS$SHU  000002     MV$IV   000001     R$$SND= 000000
MP$PRI  004411     MP$TFL  000156     MS$SRV  000007     MX$ACT  000020     R$$11M= 000000
MP$PRO  002130     MP$TH1  002206     MS$STA  000000     MX$CIR  000020     SAVALI  000256R
MP$PSS  001763     MP$TH2  002205     MS$SYN  000012     MX$CLN  000020     SAVCTX  000406R
MP$PST  001762     MP$TH3  002202     MS$TER  000001     MX$CNM  000006     SF$ANM  000001
MP$RET  001631     MP$TLN  000202     MS$TOP  000003     MX$CON  000006     SF$DNM  000002
MP$RFA  001323     MP$TLO  000172     MS$TRI  000005     MX$DAC  000020     S$$WRG= 000000
MP$RMX  002201     MP$TPA  004375     MS$UNR  000005     MX$DTE  000020     S$$YSZ= 007600
MP$RPA  004374     MP$TRI  002164     MS$VMS  000004     MX$FIL  000034     T$$KMG= 000000
MP$RPR  001606     MP$TST  000144     MS.DON  177600     MX$LGC  000777     T$$MIN= 000000
MP$RRT  002213     MP$TYP  002130     MS.MOR  000002     MX$LIN  000020     V$$CTR= 001000
MP$RST  002212     MP$UCS  004407     MS.PAR  000003     MX$LON  000006     W.BATT  000006
MP$RSV  001754     MP$USR  001750     MS.SUC  000001     MX$NMS  000454     W.BFPD  000014
MP$RSZ  001756     MP$VEC  004410     MT$AR4  000003     MX$NOD  000006     W.BHVR  000004
MP$RTI  001616     MP$VER  004406     MT$ASC  000100     MX$OBJ  000006     W.BLGH  000020
MP$RTM  001611     MP$WDF  002165     MT$BIL  000001     MX$OWN  000040     W.BLPD  000016
MP$RVE  001604     MP$WMX  002177     MT$BYS  000011     MX$PAR  000036     W.BLVR  000002
MP$RVT  000156     MP$XMT  002166     MT$CI   000007     MX$PAS  000010     W.BNPD  000015
MP$SAC  000514     MP$XPF  000242     MT$CLE  000077     MX$RAC  000047     W.BOFF  000012
MP$SAD  000543     MP$XPL  000244     MT$COD  000200     MX$RID  000047     W.BPCB  000000
MP$SCA  000310     MP$XPT  000240     MT$CON  000001     MX$RPS  000047     W.BSIZ  000010
MP$SCO  000144     MP$XXX  177777     MT$DMC  000004     MX$SID  000040     X$$DBT= 000000
MP$SCT  002176     MP$5ST  004406     MT$ETH  000006     MX$SNK  000377     $SAVRG= ****** GX
MP$SDU  000203     MS$ACT  000001     MT$HEX  000040     MX$TYP  000077     $$$SYS= 004374
MP$SDV  000160     MS$ADU  000010     MT$LPB  000005     MX$UID  000020     $$$SYX= 000000
MP$SEH  004401

. ABS.  000104     000    (RW,I,GBL,ABS,OVR)
        000550     001    (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics
```

```
                                                                           E 2
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9-4
Symbol table
```

```
                                                  .SBTTL  FMTDES - FORMAT DESTINATION PARAMETER
                                          ;+
                                          ; **-FMTDES-FORMAT DESTINATION PARAMETER
                                          ;
                                          ; THIS ROUTINE IS CALLED TO FORMAT THE ALIAS DESTINATION PARAMETER
                                          ;
                                          ; INPUTS:
                                          ;       R3 = BUFFER ADDRESS FOR PARAMETER
                                          ;       R4 = ADDRESS OF CONTEXT AREA WITH:
                                          ;               A$ALI = 6 CHARACTER ALIAS NAME (SPACE FILLED)
                                          ;               A$SCO = SCOPE OF ALIAS
                                          ;
                                          ; OUTPUTS:
                                          ;       C-CLEAR, SUCCESS
                                          ;         R3 = NEXT FREE BYTE IN BUFFER
                                          ;       C-SET, NO MATCHING ALIAS FOUND
                                          ;
                                          ; REGISTERS:
                                          ;       R4 IS PRESERVED
                                          ;-

000446                                    FMTDES:
000446  112723  000156                            MOVB    #MP$DES,(R3)+           ; SET PARAMETER TYPE = DESTINATION
000452  105023                                    CLRB    (R3)+                   ; ...
000454  112723  000304                            MOVB    #304,(R3)+              ; DATA TYPE = CODED MULT FIELDS (4)
000460  112723  000100                            MOVB    #100,(R3)+              ; DES NAME = IMAGE FIELD
                                          ;
                                          ; MAP TO SINGLE WORD POOL, AND TRY TO FIND ALIAS BLOCK
                                          ;
000464                                            SWSTK$  30$                     ;; ENTER SYSTEM STATE
000470                                            SAVMAP                          ;; SAVE CURRENT MAPPING
000474  017700  000000G                           MOV     @DECPT,R0               ;; POINT TO DECNET HOME BLOCK    ; RJK02
000500  062700  000000                            ADD     #D$ANN,R0               ;; POINT TO ALIAS NAME LISTHEAD
000504                                            CALL    FINDA                   ;; FIND THE ALIAS BLOCK
000510  103006                                    BCC     10$                     ;; IF CC, FOUND IT
000512                                            RETC    R0                      ;; RETURN CARRY TO USER STATE
000524  000431                                    BR      20$                     ;; AND RETURN
                                          ;
                                          ; SAVE DESTINATION NAME AND ACCESS CONTROL
                                          ;
000526  062701  000012                    10$:    ADD     #A.REM,R1               ;; POINT TO THE DESTINATION NAME
000532                                            CALL    FMTNAM                  ;; FORMAT THE NAME FIELD
000536                                            CALL    40$                     ;; MOVE THE USER ID STRING
000542  112723  000100                            MOVB    #100,(R3)+              ;; SET THE PASSWORD DATA TYPE
000546  112105                                    MOVB    (R1)+,R5                ;; GET THE PASSWORD LENGTH
000550  110523                                    MOVB    R5,(R3)+                ;; SET LENGTH IN BUFFER
000552  001412                                    BEQ     15$                     ;; IF EQ, THERE WAS NONE - FINISHED
000554  112763  000003  177777                    MOVB    #3,-1(R3)               ;; ELSE, SET LENGTH OF INDICATOR AS 3
000562  112723  000056                            MOVB    #'.,(R3)+               ;; SET PASSWORD INDICATIOR
000566  112723  000056                            MOVB    #'.,(R3)+               ;; ...
000572  112723  000056                            MOVB    #'.,(R3)+               ;; ...
000576  060501                                    ADD     R5,R1                   ;; SKIP OVER REMAINDER OF PASSWORD FIELD
000600                                    15$:    CALL    40$                     ;; MOVE THE ACCOUNT STRING
000604  010366  000012                            MOV     R3,10.(SP)              ;; RETURN R3 INTO SAVED R3
000610                                    20$:    RESMAP                          ;; RESTORE MAPPING
000614                                    30$:    RETURN                          ;; BACK TO USER STATE AND RETURN
```

```
     79                                          .EVEN
     80 000016      006                  ALIT3:  .BYTE   6
     81 000017      044     114     117          .ASCII  /$LOCAL/
     82                                          .EVEN
     83
     84 000026  000000'                  ALIIDX: .WORD   ALIT1
     85 000030  000010'                          .WORD   ALIT2
     86 000032  000016'                          .WORD   ALIT3
     87 000034  000000                           .WORD   0
     88
     89                                          .LIST BEX
```

```
     91                                          .SBTTL  CHKALI - CHECK ALIAS FORMAT
```

D 5

```
ALXXIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 3

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

ALCCX$           #5-58        5-65
ALUN$S           #5-57        7-285
ANBDF$           #5-57        5-60
CALL              6-127       6-141        7-217        7-278        7-283        7-288
DIR$             #7-285       7-285
HDRDF$           #5-58        5-63
MANDF$           #5-57        5-61
MAP              #5-57
MOV$             #7-285       7-285        7-285        7-285
RESMAP           #5-57
RESRG            #5-58        6-158        6-183
RETC             #5-57        7-316
RETURN            6-159       6-184        7-318        7-323        8-362
SAVMAP           #5-57
SAVRG            #5-58        6-116        6-172
SOB               6-145       6-179
SWSTK$           #5-57        7-288
UCBDF$           #5-58        5-62
```

E 5

```
**FILE**ID**ARREIN
```

```
                                                                D  6
ARREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE           REFERENCES

AF.END  = 000040        #5-65       6-170
AF.INI  = 000001        #5-65       6-203
AF.KNO  = 000002        #5-65       6-117      6-178      6-181      6-198
AF.LV2  = 000100        #5-65       6-171      6-174
AF.MUX  = 000020        #5-65
ARREIN    000000 RG     #6-88
A$BUF     000002        #5-65
A$EXA     000014        #5-65      *6-139      6-180
A$FLG     000000        #5-65      *6-117     *6-170     *6-171     *6-174      6-178     *6-181      6-198     *6-203
A$LENX    000027        #5-65       6-95
A$MAX     000006        #5-65      *6-92      *6-187     *6-188      6-200
A$NAM     000020        #5-65
A$NUM     000004        #5-65      *6-108     *6-110     *6-116     *6-180      6-200
A$OPT     000010        #5-65      *6-91       6-100      6-102
A$PAR     000016        #5-65      *6-125     *6-144     *6-190      6-213      6-214
A$XPT     000012        #5-65      *6-150
DECPT   = ******  GX     6-133
D$LNUM    000014         6-134
HEADR   = ******  GX     6-163
H.LUN     000076         6-164
I$$AS   = ******         5-61
ME$ARE    000005         6-125
ME$NOD    000000         6-144
ME.CST    177765         6-143
ME.FUN    177777         6-99
ME.IID    177767         6-124
ME.MPR    177773         6-94
ME.OPE    177747         6-191
MF$ACT    177776         6-120
MF$KNO    177777         6-118
MF$SIG    177773         6-122
MO$COU    000060         6-100      6-102
MS.MOR    000002         6-205
MXAREA  = 000077        #5-65       6-92
NF.END  = 000200         6-168
NF.LV2  = 000100         6-172
N$ROA2    000030         6-187
N$$VCT  = ******         6-193
PDVID   = ******  GX     6-142
PDVTA   = ******  GX     6-146
R$RO    = ******  GX    *6-143     *6-191
R$$MPL  = ******         6-153
R$$11D  = ******         5-61
R$$11M  = 000000         5-61
R$$11S  = ******         5-61
U.CW3     000014         6-168      6-172
X$$MCB  = ******         5-61       5-61
ZF.COU  = 001000        #5-61
ZF.DDM  = 000001        #5-61
ZF.DIA  = 004000        #5-61
ZF.DLC  = 000002        #5-61
ZF.DVP  = 100000        #5-61
```

```
                                                                E  6
ARREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00
```

```
                                                 .SBTTL  FMTCIR - FORMAT CIRCUIT ID
                                       ;+
                                       ; *** - FMTCIR  - FORMAT CIRCUIT ID
                                       ;
                                       ; INPUTS:
                                       ;          R2 = CHANNEL NUMBER
                                       ;          R3 = OUTPUT BUFFER ADDRESS
                                       ;          R4 = POINTER TO CONTEXT BLOCK
                                       ;
                                       ; OUTPUTS:
                                       ;          R3 UPDATED
                                       ;
                                       ;-

001054                                 FMTCIR:
001054  004267  000000G                          JSR     R2,$SAVVR                ; SAVE VOLATILE REGISTERS
001060  010201                                   MOV     R2,R1                    ; GET CHANNEL NUMBER
001062  003531                                   BLE     50$                      ; IF LE, NO CHANNEL
001064  012746  001466                           MOV     #MP$DLI,-(SP)            ; PARAMETER TYPE
001070  111623                                   MOVB    (SP),(R3)+               ; STORE PARAMETER TYPE
001072  116623  000001                           MOVB    1(SP),(R3)+              ; ...
001076  005726                                   TST     (SP)+                    ; CLEAN UP STACK
001100  112723  000100                           MOVB    #MT$ASC,(R3)+            ; FIELD IS ASCII IMAGE
001104  006301                                   ASL     R1                       ; CONVERT CHANNEL TO WORD INDEX
001106  105023                                   CLRB    (R3)+                    ; ASSUME NO XPT
001110  016402  000012                           MOV     A$XPT(R4),R2             ; GET XPT'S PDV ADDRESS
001114  001514                                   BEQ     50$                      ; IF EQ, NO XPT
001116  060201                                   ADD     R2,R1                    ; POINT AT CHANNEL IN PDV
001120  016146  000020                           MOV     Z.MAP(R1),-(SP)          ; GET SLN & TRIBUTARY
001124  111601                                   MOVB    (SP),R1                  ; GET SLN
001126  006301                                   ASL     R1                       ; CONVERT TO WORD INDEX
001130  067701  000000G                          ADD     @SLTMA,R1                ; POINT AT SLT ADDRESS           ; RJK03
001134  011101                                   MOV     (R1),R1                  ; POINT AT SLT


001136  010146                                   MOV     R1,-(SP)                 ; SAVE SLT ADDRESS
001140  116101  000002                           MOVB    L.DDM(R1),R1             ; GET DDM PDV INDEX
001144  067701  000000G                          ADD     @PDVTA,R1                ; POINT AT PDV ADDRESS           ; RJK03
001150  011101                                   MOV     (R1),R1                  ; POINT AT DDM PDV
001152  042764  000020  000000                   BIC     #AF.MUX,A$FLG(R4)        ; CLEAR MUX BIT IN CONTEXT AREA
001160  032761  000040  000010                   BIT     #ZF.MUX,Z.FLG(R1)        ; IS THIS A MUX DEVICE ?
001166  001403                                   BEQ     10$                      ; IF EQ, NO
001170  052764  000020  000000                   BIS     #AF.MUX,A$FLG(R4)        ; ELSE, REMEMBER IT
001176  016101  000004            10$:           MOV     Z.NAM(R1),R1             ; GET DDM PROCESS NAME
001202  010300                                   MOV     R3,R0                    ; COPY BUFFER ADDRESS
001204                                           CALL    $C5TA                    ; CONVERT TO ASCII
001210  124027  000040            20$:           CMPB    -(R0),#'                 ; DELETE TRAILING SPACES
001214  001775                                   BEQ     20$                      ; ...
001216  005200                                   INC     R0                       ; WENT TOO FAR
001220  112720  000055                           MOVB    #'-,(R0)+                ; INSERT DELIMETER
001224  011601                                   MOV     (SP),R1                  ; GET SLT ADDRESS
001226  116101  000012                           MOVB    L.CTL(R1),R1             ; GET CONTROLLER NUMBER
001232  012702  014012                           MOV     #<<3*4000>!10.>,R2       ; USE BASE 10. FIELD WIDTH = 3
001236                                           CALL    $CBTA                    ; CONVERT TO ASCII
001242  032764  000020  000000                   BIT     #AF.MUX,A$FLG(R4)        ; IS THIS A MUX DEVICE ?
001250  001411                                   BEQ     30$                      ; IF EQ, NO
001252  112720  000055                           MOVB    #'-,(R0)+                ; ELSE, INSERT DELIMETER
```

```
001256  011601                                   MOV     (SP),R1                  ; GET SLT ADDRESS
```

```
                                                                         D 8
**FILE**ID**BLNLST









BBBBBBBB     LL           NN      NN   LL             SSSSSSSS   TTTTTTTTTT
BBBBBBBB     LL           NN      NN   LL             SSSSSSSS   TTTTTTTTTT
BB      BB   LL           NN      NN   LL           SS               TT
BB      BB   LL           NN      NN   LL           SS               TT
BB      BB   LL           NNNN    NN   LL           SS               TT
BB      BB   LL           NNNN    NN   LL           SS               TT
BBBBBBBB     LL           NN  NN  NN   LL             SSSSSS         TT
BBBBBBBB     LL           NN  NN  NN   LL             SSSSSS         TT
BB      BB   LL           NN    NNNN   LL                   SS       TT
BB      BB   LL           NN    NNNN   LL                   SS       TT
BB      BB   LL           NN      NN   LL                   SS       TT           ....
BB      BB   LL           NN      NN   LL                   SS       TT           ....
BBBBBBBB     LLLLLLLLLL   NN      NN   LLLLLLLLLL   SSSSSSSS         TT           ....
BBBBBBBB     LLLLLLLLLL   NN      NN   LLLLLLLLLL   SSSSSSSS         TT           ....








LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                         E 8
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08
Table of contents
```

```
    286                                          .SBTTL  SAVECL - SAVE ECL NODES
    287                                  ;+
    288                                  ; *** - SAVECL - SAVE ECL NODES
    289                                  ;
    290                                  ; INPUTS:
    291                                  ;        R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
    292                                  ;
    293                                  ; OUTPUTS:
    294                                  ;        R5 = UPDATED
    295                                  ;
    296                                  ;-
    297
    298 000542                           SAVECL: MAPECL                          ;; MAP/POINT TO ECL DATABASE
    299 000546  062702  000042                   ADD     #N$ENC,R2               ;; GET POINTER TO ECL NODE DATA BASE
    300 000552                                   MAP     4(R2)                   ;; GET APR BIAS TO MAP TO
    301
    302 000560  011202                   10$:    MOV     (R2),R2                 ;; GET NEXT ENTRY
    303 000562  001420                           BEQ     30$                     ;; IF EQ, END OF LIST
    304
    305 000564  005762  000002                   TST     E$NNOD(R2)              ;; WAS THIS COUNTER TABLE USED ?
    306 000570  001773                           BEQ     10$                     ;; IF EQ, NO .. KEEP LOOKING
    307 000572  016201  000002                   MOV     E$NNOD(R2),R1           ;; GET NODE ADDRESS
    308 000576  020164  000036                   CMP     R1,N$HECL(R4)           ;; IS ADDRESS HIGHER THAN ECL HIGHEST
    309 000602  101402                           BLOS    20$                     ;; IF LOS, NO - BRANCH
    310 000604  010164  000036                   MOV     R1,N$HECL(R4)           ;; ELSE SET HIGHEST ECL TO CURRENT ADDRESS
    311 000610  020164  000046           20$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
    312 000614  101761                           BLOS    10$                     ;; IF LOS, NO - RETURN
    313 000616                                   CALL    SAVCTX                  ;; SAVE THIS NODE ADDRESS
    314 000622  000756                           BR      10$                     ;; KEEP LOOKING
    315
    316 000624                           30$:    RETURN
```

```
    318                                          .SBTTL  SAVLOO - SAVE LOOP NODE ADDRESSES
```

D 10

```
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09       PAGE 2
SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL   VALUE             REFERENCES

E$NTIM     000046          #5-68
E$NUSE     000004          #5-68
E$STRT     000006          #5-68
I$$AS   =  ******           5-69
KISAR6  =  ******   GX      8-131      *8-133      *10-206     *12-300     *14-388
LF$END  =  000020          #6-87
LF$MLT  =  040000          #6-87
LF$REA  =  000001          #6-87
LF$SEG  =  100000          #6-87
LF$SIG  =  000040          #6-87
LF$SKP  =  000004          #6-87
LF$VR2  =  000010          #6-87
LF$ZER  =  000002          #6-87
LP$MPT  =  000010          #6-87
LP$MUX  =  000004          #6-87
LP$TRB  =  000002          #6-87
LP$UNT  =  000001          #6-87
LP$WCN  =  000040          #6-87       6-87
LP$WDV  =  000020          #6-87       6-87
LP$WLD  =  000360          #6-87
LP$WTR  =  000200          #6-87       6-87
LP$WUN  =  000100          #6-87       6-87
L$ADJ      000110          #5-87
L$BUF      000060          #5-87
L$CHN      000054          #5-87
L$COU      000114          #5-87
L$CTB      000053          #5-87
L$CTL      000002          #5-87
L$DDM      000000          #5-87
L$FLG      000064          #5-87
L$FLX      000070          #5-87
L$LEN      000124          #5-87
L$LTM      000070          #5-87
L$MSC      000056          #5-87
L$MTYP     000067          #5-87
L$NAM      000000          #5-87
L$NLEN  =  000020          #5-87       5-87        5-87
L$NOD      000112          #5-87
L$NXT      000044          #5-87
L$OPT      000062          #5-87
L$PAR      000120          #5-87
L$PDV      000055          #5-87
L$PFG      000041          #5-87
L$PLB      000116          #5-87
L$PRO      000067          #5-87
L$PVC      000004          #5-87
L$SCN      000020          #5-87
L$SCR      000072          #5-87
L$SLEN     000060          #5-87
L$SLT      000042          #5-87
L$SNM      00004           #5-87
L$SYL      00012           #5-87
```

E 10

```
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09       PAGE 3
SYMBOL CROSS REFERENCE                                      CREF   04.00
```

```
                                                       D 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-1
Symbol table

A$$CHK=  000000          LF$VR2=  000010          MB$TXO   000000          MC$P11   000001          MD$DPV   000044
A$$CPS=  000000          LF$ZER=  000002          MB$ZER   000000          MC$RFL   002046          MD$DQ    000006
A$$PRI=  000000          LP$MPT=  000010          MC$BAB   006073          MC$SBU   002051          MD$DTE   000024
A$$TRP=  000000          LP$MUX=  000004          MC$BID   001765          MC$SFL   002044          MD$DU    000002
BIT0  =  000001          LP$TRB=  000002          MC$BMC   001767          MC$SYC   000012          MD$DUP   000012
BIT1  =  000002          LP$UNT=  000001          MC$BSC   001766          MC$SYL   000036          MD$DV    000026
BIT10 =  002000          LP$WCN=  000040          MC$CAC   001442          MC$SYR   000050          MD$DZ    000030
BIT11 =  004000          LP$WDV=  000020          MC$CAP   001440          MC$SYS   000024          MD$FUL   000000
BIT12 =  010000          LP$WLD=  000360          MC$CCL   001445          MC$SYZ   000000          MD$HAL   000001
BIT13 =  020000          LP$WTR=  000200          MC$CDC   002045          MC$TBR   001140          MD$HEL   000002
BIT14 =  040000          LP$WUN=  000100          MC$CDP   001441          MC$TBS   001141          MD$KCP   000013
BIT15 =  100000          L$ADJ    000110          MC$CIF   001465          MC$TDR   002053          MD$KDP   000034
BIT2  =  000004          L$BUF    000060          MC$CLD   001464          MC$TIM   000000          MD$KDZ   000036
BIT3  =  000010          L$CHN    000054          MC$COU   100000          MC$TYP   007777          MD$KL    000040
BIT4  =  000020          L$COU    000114          MC$CTL   001454          MC$UBU   002052          MD$KMX   000054
BIT5  =  000040          L$CTB    000053          MC$CTR   001452          MC$UFD   002047          MD$KMY   000052
BIT6  =  000100          L$CTL    000002          MC$CTS   001453          MC$UMR   001132          MD$LEN   000050
BIT7  =  000200          L$DDM    000000          MC$DOV   002050          MC$UMS   001133          MD$PCL   000011
BIT8  =  000400          L$FLG    000064          MC$LBR   001750          MC$VAX   000003          MD$QNA   000005
BIT9  =  001000          L$FLX    000070          MC$LBS   001751          MC$WID   060000          MD$UNA   000001
B$10     000003          L$LEN    000124          MC$LDI   001774          MC$WIH   040000          MD$WIT   000002
B$11     000004          L$LTM    000070          MC$LDO   001775          MC$WIL   020000          MD.ABO   000017
B$12     000005          L$MSG    000056          MC$LDR   001762          MC$W08   020000          MD.ACC   000010
B$5      000000          L$MTYP   000067          MC$LDS   001763          MC$W16   040000          MD.AOB   000016
B$6      000001          L$NAM    000000          MC$LLB   002021          MC$W32   060000          MD.BOB   000011
B$7      000002          L$NLEN=  000020          MC$LLP   002115          MC$XBR   001750          MD.DIA   000007
B$7S10   000006          L$NOD    000112          MC$LLR   002007          MC$XBS   001751          MD.DOB   000015
CHKCIR   000000RG        L$NXT    000044          MC$LUO   000001          MC$XCJ   000322          MD.DUM   000002
C$$CKP=  000000          L$OPT    000062          MC$LRB   002020          MC$XCR   002260          MD.FAI   000014
C$$ORE=  000400          L$PAR    000120          MC$LRP   002114          MC$XCS   002261          MD.FNA   000001
C$$RSH=  177564          L$PDV    000055          MC$LRR   002006          MC$XDR   001762          MD.FOB   000006
D$$BUG=  177514          L$PFG    000041          MC$LSI   002032          MC$XDS   001763          MD.LOA   000001
D$$ISK=  000000          L$PLB    000116          MC$LST   002033          MC$XFR   002272          MD.LSH   000020
D$$L11=  000001          L$PRO    000067          MC$LUP   006072          MC$XFS   002273          MD.NOB   000012
D$$YNC=  000000          L$PVC    000004          MC$MAP   010000          MC$XLJ   000323          MD.NON   177777
D$$YNM=  000000          L$SCN    000020          MC$MBL   001764          MC$XLR   002330          MD.PER   000000
E$$XPR=  000000          L$SCR    000072          MC$MBX   004420          MC$XMA   000310          MD.RES   000004
F$$LVL=  000001          L$SLEN   000066          MC$MBY   001752          MC$XMC   002305          MD.ROB   000005
G$$TPP=  000000          L$SLT    000042          MC$MPX   004406          MC$XMS   002304          MD.RSH   000013
G$$TSS=  000000          L$SNM    000046          MC$NAP   001604          MC$XNR   002332          MD.SDU   000005
G$$TTK=  000000          L$SYL    000122          MC$NBR   001130          MC$XRC   002316          MD.SLO   000003
G$$WRD=  000000          L$TNM    000052          MC$NBS   001131          MC$XRR   002331          MD.SNA   000000
I$$RAR=  000000          L$TPT    000050          MC$NCR   001154          MC$XRS   002342          MD.TLO   000004
I$$RDN=  000000          L$TRB    000040          MC$NCS   001155          MC$020   000002          MD.UNA   000002
K$$CNT=  177546          L$TSZ    000070          MC$NML   001274          MD$CJ    000007          MD.UNR   000003
K$$CSR=  177546          L$TYP    000063          MC$NMR   001142          MD$CNA   000003          MD.UOB   000007
K$$LDC=  000000          L$UNT    000003          MC$NMS   001143          MD$COU   000001          MD.VOL   000006
K$$TPS=  000074          L$$ASG=  000000          MC$NNO   001606          MD$DA    000010          ME$ALI   000010
LC$EXT   000000          L$$DRV=  000000          MC$NNU   001605          MD$DL    000004          ME$AL2   000003
LC$INT   000001          L$$P11=  000001          MC$NOP   001607          MD$DLV   000020          ME$ARE   000005
LD$LF =  000000          L$$11R=  000000          MC$NOR   000000          MD$DMC   000014          ME$CIR   000003
LF$END=  000020          MB$DIS   000001          MC$NPF   001616          MD$DMF   000046          ME$EXA   000000
LF$MLT=  040000          MB$ENA   000000          MC$NPR   001630          MD$DMP   000022          ME$EXE   000200
LF$REA=  000001          MB$FUL   000002          MC$NRE   001200          MD$DMR   000050          ME$LIN   000001
LF$SEG=  100000          MB$MIX   000002          MC$NRT   001166          MD$DMV   000042          ME$LOG   000002
LF$SIG=  000040          MB$ONE   000001          MC$NVR   001642          MD$DN    000016          ME$MOD   000004
LF$SKP=  000004          MB$RXC   000001          MC$P08   000000          MD$DP    000000          ME$NOD   000000
```

```
                                                       E 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturuay 29-Jun-85 17:41  Page 7-2
Symbol table
ME$NON   177777          ML$EXI   000001          MP$PRI   002165          MP$DUC   000210          MP$MDE   004421
```

```
                                                     .SBTTL  CHKLIN - CHECK LINE ID FORMAT
                                             ;+
                                             ; **-CHKLIN-CHECK LINE ID FORMAT
                                             ;
                                             ; THIS ROUTINE CHECKS THE PASSED LINE ID FORMAT FOR VALIDITY,
                                             ; AND IF VALID, SAVES THE LINE ID CONTEXT.
                                             ;
                                             ; INPUTS:
                                             ;       R1 = LINE ID ADDRESS
                                             ;       R4 = ADDRESS OF CONTEXT AREA
                                             ;
                                             ; OUTPUTS:
                                             ;       C-BIT ERROR INDICATOR
                                             ;         IF CLEAR, L$TYP(R4) = SAVED LINE ID TYPE AND:
                                             ;            IF TYPE >0 , LINE ID PARSED
                                             ;         IF SET, ERROR R0 CONTAINS ERROR CODE
                                             ;
                                             ; REGISTERS:
                                             ;       ALL REGISTERS ARE PRESERVED UNLESS THERE IS AN ERROR (SEE ABOVE)
                                             ;-

000000                                       CHKLIN::
                                             ;
                                             ; CHECK LINE FORMAT TYPE
                                             ;
000000  121127  177773                               CMPB    (R1),#MF$SIG            ; SHOW SIGNIFIGANT LINES ?
000004  001403                                       BEQ     5$                      ; BR IF YES
000006  121127  177776                               CMPB    (R1),#MF$ACT            ; IS IT WITHIN VALID RANGE ?
000012  002414                                       BLT     20$                     ; IF LT, NO - ERROR
000014  152764  000161  000041       5$:             BISB    #LP$WDV!LP$WCN!LP$WUN!LP$UNT,L$PFG(R4)
                                                                                     ; ASSUME ACT OR KNOWN LINES (WILD EVERYTHING)
000022  111164  000063                               MOVB    (R1),L$TYP(R4)          ; SAVE LINE FORMAT TYPE
000026  002404                                       BLT     10$                     ; IF LT, NO LINE NAME - RETURN
000030  001405                                       BEQ     20$                     ; IF EQ, INVALID ID - ERROR
                                             ;
                                             ; PARSE THE LINE-ID STRING
                                             ;
000032                                               CALL    PRSLIN                  ; PARSE THE LINE-ID
000036  000405                                       BR      30$                     ; AND RETURN
000040  000241                       10$:            CLC                             ; SET SUCCESS
000042  000403                                       BR      30$                     ; AND RETURN
000044  012700  177767               20$:            MOV     #ME.IID,R0              ; INVALID LINE-ID
000050  000261                                       SEC                             ; ...
000052                               30$:            RETURN

        000001                                       .END
        000125                             L$INA:  .BLKB   1               ; DMP ACTIVE INCREMENT
        000126                             L$IND:  .BLKB   1               ; DMP DYING INCREMENT
        000127                             L$INI:  .BLKB   1               ; DMP INACTIVE INCREMENT
        000130                             L$TH1:  .BLKB   1               ; DMP DEAD THRESHOLD
        000131                             L$TH2:  .BLKB   1               ; DMP DYING THRESHOLD
        000132                             L$TH3:  .BLKB   1               ; DMP INACTIVE THRESHOLD
        000133                             L$MXB:  .BLKB   1               ; DMP MAXIMUM BLOCKS
        000134                             L$NTL:  .BLKW   13.             ; NTL MESSAGE BUFFER
        000166                             L$SCR:  .BLKW   15.             ; SCRATCH BUFFER
        000224                             L$LCT:  .BLKW   1               ; COUNTER TIMER
        000226                             L$HTM:  .BLKW   1               ; HELLO TIMER
        000230                             L$SER:                          ; SERVICE
        000230                             L$LTM:  .BLKW   1               ; LISTEN TIMER
        000232                             L$XCH:  .BLKW   1               ; X25 Logical Channel Number (LCN)
        000234                             L$LMB:                          ; X25 Max Block (Line)
        000234                             L$CMB:  .BLKW   1               ; X25 Max Data (Circuit)
        000236                             L$NUML:                         ; DLM Number length
        000236                             L$DTEL: .BLKW   1               ; X25 DTE length (Circuit)
        000240                             L$NUM:                          ; DLM Number
        000240                             L$DTE:  .BLKB   8.              ; X25 DTE (Circuit)
        000250                             L$DTEP: .BLKW   1               ; X25 Pointer to DTE descriptor (Circuit)
        000252                             L$MWN:                          ; X25 Max Window (Line)
        000252                             L$MXW:  .BLKB   1               ; X25 Max Window (Circuit)
        000253                             L$MRT:                          ; X25 Max Retransmits (Line)
        000253                             L$MXR:  .BLKB   1               ; DLM Max Recalls (Circuit)
        000254                             L$HBT:  .BLKW   1               ; X25 Holdback Timer (Line)
        000256                             L$NTI:                          ; X25 Retransmit Timer (Line)
        000256                             L$RET:  .BLKW   1               ; DLM Recall Timer (Circuit)
        000260                             L$CUS:  .BLKW   1               ; DLM Usage
        000262                             L$BLK:  .BLKW   1               ; DLM blocking state
                                                   .IF DF  R$$PRO  ;PRO/DECnet
                                           L$LTY:  .BLKW   1               ; Loopback Type
                                           L$MDT:  .BLKW   1               ; Modem Test
                                                   .ENDC   ; DF R$$PRO
                                                   .EVEN
        000264                             L$LEN:                          ; LENGTH OF CONTEXT AREA
        000000                                     .PSECT

                                           ;
                                           ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                           ;
                000001                             LF$REA = 1              ; READ COUNTERS OPERATION
                000002                             LF$ZER = 2              ; ZERO COUNTERS OPERATION
                000004                             LF$SKP = 4              ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                           ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                           ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                           ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                           ; OF THE TRIBUTARY COUNTERS.
                000010                             LF$VR2 = 10             ; CONNECTED TO VERSION 2.0 NCP
                040000                             LF$MLT  = 40000         ; MULTIPLE ADJACENCY FLAG
                100000                             LF$SEG  = 100000        ; SEGMENTED RESPONSE IN PROGRESS

                                           ;
                                           ; PARSE FLAG DEFINITIONS (L$PFG)
                                           ;
                000001                             LP$UNT = 1              ; UNIT NUMBER FOUND
                000002                             LP$TRB = 2              ; TRIBUTARY NUMBER FOUND
                000004                             LP$MUX = 4              ; DEVICE IS MUX
                000010                             LP$MPT = 10             ; LINE IS MULTIPOINT
```

```
                000020                             LP$WDV = 20             ; WILD CARD DEVICE NAME FOUND
```

```
                                                .SBTTL  CICHIN - CIRCUIT CHANGE PARAMETERS INITIALIZE
                                        ;+
                                        ; **-CICHIN-CIRCUIT CHANGE PARAMETERS INITIALIZE ROUTINE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO INITIALIZE FOR THE CIRCUIT CHANGE PARAMETER
                                        ; OPERATION.
                                        ;
                                        ; INPUTS:
                                        ;       R0 = OPTION (SET/CLEAR)
                                        ;       R1 = ID ADDRESS (LINE FORMAT)
                                        ;       R2 = LENGTH OF DATA IN INPUT BUFFER
                                        ;       R3 = ADDRESS OF INPUT BUFFER
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = RETURN CODE
                                        ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                        ;       R3 = ADDESS OF OUTPUT BUFFER
                                        ;       DATA SAVED IN CONTEXT AREA
                                        ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                        ;
                                        ; REGISTERS:
                                        ;       R3, R4, R5 ARE PRESERVED
                                        ;-

000000                                  CICHIN::
000000  004567  000000G                         JSR     R5,$SAVRG               ; SAVE R3-R5
                                        ;
                                        ; INITIALIZE CONTEXT AREA
                                        ;
000004  110064  000062                          MOVB    R0,L$OPT(R4)            ; SAVE OPTION BYTE
000010  010364  000060                          MOV     R3,L$BUF(R4)            ; SAVE BUFFER POINTER
                                        ;
                                        ; CHECK OPTIONS FLAGS
                                        ;
000014  012700  177777                          MOV     #ME.FUN,R0              ; ASSUME ILLEGAL OPTION
000020  132764  177677  000062                  BITB    #^C<MO$CLE>,L$OPT(R4)   ; CHECK FOR ILLEGAL OPTIONS
000026  105764  000062                          TSTB    L$OPT(R4)               ; CHECK FOR ILLEGAL OPTIONS
                                                BNE     30$                     ; IF NE, ERROR
                                        ;
                                        ; CHECK AND SAVE LINE FORMAT
                                        ;
000032  012764  000003  000072                  MOV     #ME$CIR,L$PAR(R4)       ; SET ERROR DETAIL
000040                                          CALL    CHKCIR                  ; CHECK CIRCUIT FORMAT
000044  103426                                  BCS     30$                     ; Branch if invalid circuit
000046  012700  177767                          MOV     #ME.IID,R0              ; ASSUME INVALID LINE-ID
000052  126427  000063  177776                  CMPB    L$TYP(R4),#MF$ACT       ; ELSE, WAS IT ACTIVE ?
000060  001420                                  BEQ     30$                     ; IF EQ, YES - DOESN'T APPLY HERE
                                        ;
                                        ; PARSE MESSAGE AND SAVE PARAMETERS
                                        ;
000062  005064  000072                  10$:    CLR     L$PAR(R4)               ; INITIALIZE ERROR DETAIL
000066  012700  177776                          MOV     #ME.FOR,R0              ; ASSUME MESSAGE FORMAT ERROR
000072  012705  000000'                         MOV     #SETCIR,R5              ; POINT TO CIRCUIT MESSAGE STATE TABLE
000076  105764  000062                          TSTB    L$OPT(R4)               ; WAS OPTION A SET ?
000102  001402                                  BEQ     20$                     ; IF EQ, YES - PROPER TABLE
                                          .IF NDF R$$11S
```

```
                                         .SBTTL  CMBACT - X.25 Circuit maximum data action routine
                                 ;+
                                 ; *** CMBACT - X.25 Circuit maximum data action routine
                                 ;
                                 ; INPUTS
                                 ;        .MSGPT   - Pointer to maximum data parameter value
                                 ;        R4       - Pointer to context area
                                 ;
                                 ; OUTPUTS
                                 ;        L$CMB(R4) Contains maximum data parameter
                                 ;-

000710                           CMBACT:
000710  016701  000000G                  MOV     .MSGPT,R1               ; Get pointer to data
000714  112164  000234                   MOVB    (R1)+,L$CMB(R4)         ; Get low byte of data
000720  111164  000235                   MOVB    (R1),L$CMB+1(R4)        ; Get high byte of data
000724                                   RETURN
```

```
   1164                                                 .SBTTL  BSIACT - INACTIVE BASE ACTION ROUTINE
   1165
   1166                                         ;+
   1167                                         ;
   1168                                         ; BSIACT - INACTIVE BASE ACTION ROUTINE
   1169                                         ;
   1170                                         ; INPUTS:
   1171                                         ;       .MSGPT - POINTER TO 1 BYTE INACTIVE BASE
   1172                                         ;       R4     - POINTER TO CONTEXT AREA
   1173                                         ;
   1174                                         ; OUTPUTS:
   1175                                         ;       L$BSI(R4) - INACTIVE BASE VALUE
   1176                                         ;
   1177                                         ;-
   1178 001272                                  BSIACT: CALL    CHKDMP                  ; CHECK FOR DMP
   1179 001276  110164  000124                          MOVB    R1,L$BSI(R4)            ; SAVE INACTIVE BASE VALUE
   1180 001302                                          RETURN
```

```
**FILE**ID**ALREIN
```

```
  AAAAAA     LL           RRRRRRRR     EEEEEEEEEE     IIIIII     NN      NN
  AAAAAA     LL           RRRRRRRR     EEEEEEEEEE     IIIIII     NN      NN
AA      AA   LL           RR      RR   EE               II       NN      NN
AA      AA   LL           RR      RR   EE               II       NN      NN
AA      AA   LL           RR      RR   EE               II       NNNN    NN
AA      AA   LL           RR      RR   EE               II       NNNN    NN
AA      AA   LL           RRRRRRRR     EEEEEEEE         II       NN  NN  NN
AA      AA   LL           RRRRRRRR     EEEEEEEE         II       NN  NN  NN
AAAAAAAAAA   LL           RR  RR       EE               II       NN    NNNN
AAAAAAAAAA   LL           RR  RR       EE               II       NN    NNNN
AA      AA   LL           RR    RR     EE               II       NN      NN         ....
AA      AA   LL           RR    RR     EE               II       NN      NN         ....
AA      AA   LLLLLLLLLL   RR      RR   EEEEEEEEEE     IIIIII     NN      NN         ....
AA      AA   LLLLLLLLLL   RR      RR   EEEEEEEEEE     IIIIII     NN      NN         ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05
Table of contents
```

```
Symbol table


Work  file  reads: 208
Work  file writes: 141
Size of work file: 27775 Words  ( 109 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:27.11
SY:ALREIN.V2,[135,134]ALREIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ALREIN
```

```
SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
                                        ;
                                        ; LOCAL SUBROUTINE FOR FORMATTING ACCESS CONTROL STRINGS
                                        ;
 000616  112723  000100                 40$:    MOVB    #100,(R3)+              ;; DATA TYPE = IMAGE FIELD
 000622  112105                                 MOVB    (R1)+,R5                ;; GET STRING LENGTH
 000624  110523                                 MOVB    R5,(R3)+                ;; SET STRING LENGTH IN BUFFER
 000626  005305                         50$:    DEC     R5                      ;; LOOP FOR STRING LENGTH
 000630  002402                                 BLT     60$                     ;; ...
 000632  112123                                 MOVB    (R1)+,(R3)+             ;; MOVE STRING INTO BUFFER
 000634  000774                                 BR      50$                     ;; ...
 000636                                 60$:    RETURN
```

```
                                                .SBTTL  FINDA - FIND A MATCHING ALIAS BLOCK
```

```
                                        .SBTTL  CHKALI - CHECK ALIAS FORMAT
                                ;+
                                ; **-CHKALI-CHECK ALIAS FORMAT
                                ;
                                ; THIS ROUTINE CHECKS THE PASSED ALIAS FORMAT FOR VALIDITY,
                                ; AND IF VALID, SAVES THE ALIAS IN THE CONTEXT AREA.
                                ;
                                ; INPUTS:
                                ;       R1 = ID ADDRESS (ALIAS FORMAT)
                                ;       R4 = ADDRESS OF CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       C-BIT ERROR INDICATOR
                                ;         IF CLEAR, A$TYP(R4) = SAVED ALIAS TYPE, AND A$ALI(R4) = ALIAS NAME
                                ;              SPACE FILLED, IF TYPE > 0
                                ;         IF SET, ERROR RETURN IN R0
                                ;
                                ; REGISTERS:
                                ;       R2, R3, R4 ARE PRESERVED
                                ;-

 000036                         CHKALI::
                                ;
                                ; CHECK ALIAS FORMAT TYPE
                                ;
 000036                                 SAVRG   R2                      ; MAKE WORKSPACE
 000040 121127  177775                  CMPB    (R1),#MF$ALL            ; IS IT WITHIN VALID RANGE ?
 000044 002436                          BLT     20$                     ; IF LT, NO - ERROR
 000046 111164  000011                  MOVB    (R1),A$TYP(R4)          ; ELSE, SAVE ALIAS FORMAT TYPE
 000052 002453                          BLT     30$                     ; IF LT, NO ALIAS NAME - RETURN
                                ;
                                ; IF FORMAT > 0 , SAVE ALIAS NAME AND CHECK FOR SYNTAX
                                ;
 000054 010400                          MOV     R4,R0                   ; COPY CONTEXT AREA POINTER
 000056 062700  000012                  ADD     #A$ALI,R0               ; POINT TO ALIAS NAME BUFFER
 000062 010146                          MOV     R1,-(SP)                ; SAVE POINTER TO ALIAS DESCRIPTOR
 000064                                 CALL    SAVNAM                  ; CHECK THE NODE NAME AND SAVE IT
 000070 012601                          MOV     (SP)+,R1                ; RESTORE ALIAS DESCRIPTOR POINTER
 000072 103043                          BCC     30$                     ; IF CC, GOOD NAME - RETURN
                                ;
                                ; SPECIAL CASE CHECKS OF ALIAS NAMES , ALL BLANKS, $HOST AND $LOCAL
                                ;
 000074 112105                          MOVB    (R1)+,R5                ; GET LENGTH OF NAME
 000076 001421                          BEQ     20$                     ; NULL STRING, KICK IT OUT
 000100 120527  000006                  CMPB    R5,#6                   ; WAS THE NAME TOO LONG ?
 000104 003016                          BGT     20$                     ; IF GT, YES - ERROR
 000106 010046                          MOV     R0,-(SP)                ; SAVE POINTER TO ALIAS NAME BUFFER
 000110 012702  000026'                 MOV     #ALIIDX,R2              ; POINT TO NAME TABLE INDEX
 000114 012200                  10$:    MOV     (R2)+,R0                ; POINT TO A NAME
 000116 001410                          BEQ     17$                     ; IF EQ, DONE AND NO MATCH
 000120                                 CALL    CMPCST                  ; COMPARE INPUT NAME TO TABLE
 000124 103773                          BCS     10$                     ; NO MATCH, TRY AGAIN
 000126 012600                          MOV     (SP)+,R0                ; MATCH, RESTORE PTR TO BUFFER
 000130 112120                  15$:    MOVB    (R1)+,(R0)+             ; AND COPY THE NAME ...
 000132                                 SOB     R5,15$                  ; INTO THE ALIAS NAME BUFFER
 000136 000421                          BR      30$                     ; CLEAN UP AND CONTINUE
 000140 005726                  17$:    TST     (SP)+                   ; CLEAN STACK
```

```
**FILE**ID**ARREIN




  AAAAAA     RRRRRRRR     RRRRRRRR     EEEEEEEEEE     IIIIII     NN      NN
  AAAAAA     RRRRRRRR     RRRRRRRR     EEEEEEEEEE     IIIIII     NN      NN
AA      AA   RR      RR   RR      RR   EE               II       NN      NN
AA      AA   RR      RR   RR      RR   EE               II       NN      NN
AA      AA   RR      RR   RR      RR   EE               II       NNNN    NN
AA      AA   RR      RR   RR      RR   EE               II       NNNN    NN
AA      AA   RRRRRRRR     RRRRRRRR     EEEEEEEE         II       NN  NN  NN
AA      AA   RRRRRRRR     RRRRRRRR     EEEEEEEE         II       NN  NN  NN
AAAAAAAAAA   RR  RR       RR  RR       EE               II       NN    NNNN
AAAAAAAAAA   RR  RR       RR  RR       EE               II       NN    NNNN
AA      AA   RR    RR     RR    RR     EE               II       NN      NN       ....
AA      AA   RR    RR     RR    RR     EE               II       NN      NN       ....
AA      AA   RR      RR   RR      RR   EEEEEEEEEE     IIIIII     NN      NN       ....
AA      AA   RR      RR   RR      RR   EEEEEEEEEE     IIIIII     NN      NN       ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07
Table of contents
```

```
                                                                 E  6
ARREIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:07      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE               REFERENCES

ZF.INI  = 040000            #5-61
ZF.KMX  = 000020            #5-61
ZF.LLC  = 000004            #5-61
ZF.LMC  = 000100            #5-61
ZF.MAN  = 020000            #5-61
ZF.MFL  = 000010            #5-61
ZF.MTM  = 000400            #5-61
ZF.MUX  = 000040            #5-61
ZF.PSE  = 002000            #5-61
ZF.SLI  = 010000            #5-61
ZF.TIM  = 000200            #5-61
ZF.X3P  = 000000            #5-61
ZS.ASN  = 100000            #5-61
ZS.BSY  = 140000            #5-61
Z.AVL     000014            #5-61
Z.DAT     000016            #5-61       6-151
Z.DSP     000000            #5-61       5-61        6-148
Z.FLG     000010            #5-61
Z.LEN   = 000016            #5-61
Z.LLN     000006            #5-61
Z.MAP     000020            #5-61
Z.NAM     000004            #5-61
Z.PCB     000012            #5-61
Z.SCH     000007            #5-61
$HEADR  = ******    GX       6-193
$MBLUN  = ******    GX       6-164
$SAVRG  = ******    GX       6-89
```

```
                                                                 F  6
ARREIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:07      PAGE 3
MACRO CROSS REFERENCE                                     CREF    04.00
```

```
    379 001256  011601                              MOV     (SP),R1                 ; GET SLT ADDRESS
    380 001260  116101  000013                      MOVB    L.UNT(R1),R1            ; GET UNIT NUMBER
    381 001264  012702  014012                      MOV     #<<3*4000>!10.>,R2      ; USE BASE 10. FIELD WIDTH = 3
    382 001270                                      CALL    $CBTA                   ; CONVERT TO ASCII
    383 001274  012601                      30$:    MOV     (SP)+,R1                ; GET SLT ADDRESS
    384 001276  032761  000020  000000              BIT     #LF.MTP,L.FLG(R1)       ; IS THIS LINE MULTI-POINT ?
    385 001304  001413                              BEQ     40$                     ; IF EQ, NO
    386 001306  105761  000014                      TSTB    L.NSTA(R1)              ; MULTIPOINT SLAVE?
    387 001312  001410                              BEQ     40$                     ; BR IF YES
    388 001314  112720  000056                      MOVB    #'.,(R0)+               ; ELSE, INSERT DELIMETER
    389 001320  116601  000001                      MOVB    1(SP),R1                ; GET TRIBUTARY ADDRESS
    390 001324  012702  014012                      MOV     #<<3*4000>!10.>,R2      ; USE BASE 10. FIELD WIDTH = 3
    391 001330                                      CALL    $CBTA                   ; CONVERT TO ASCII
    392 001334  005726                      40$:    TST     (SP)+                   ; CLEAN UP STACK
    393 001336  160300                              SUB     R3,R0                   ; CALCULATE STRING LENGTH
    394 001340  110063  177777                      MOVB    R0,-1(R3)               ; SET LENGTH OF IMAGE FIELD
    395 001344  060003                              ADD     R0,R3                   ; POINT PAST STRING
    396 001346                              50$:    RETURN
    397
    398
```

Table of contents

```
     5-   47    MACRO CALLS AND LOCAL DATA
     6-   87    BIT DEFINITIONS
     7-   94    DISPATCH TABLE FOR SAVE ROUTINES
     8-  114    RBNLST - REBUILD NODE LIST IN CONTEXT AREA
     9-  137    BLNLST - BUILD NODE LIST IN CONTEXT AREA
    10-  189    SAVROL - SAVE ROL NODE ADDRESSES
    11-  246    SAVREM - SAVE REMOTE NODE ADDRESSES
    12-  286    SAVECL - SAVE ECL NODES
    13-  318    SAVLOO - SAVE LOOP NODE ADDRESSES
    14-  370    SAVADJ - SAVE ADJACENT NODE ADDRESSES
    15-  428    SAVSER - SAVE SERVICE NODE ADDRESS
    16-  459    SAVCTX - SAVE NODE CONTEXT
```

```
      1                                     .TITLE  BLNLST - BUILD NODE LIST IN CONTEXT AREA
```

```
                                        .SBTTL  SAVLOO - SAVE LOOP NODE ADDRESSES
                                ;+
                                ; *** - SAVLOO - SAVE LOOP NODE ADDRESSES
                                ;
                                ; THIS ROUTINE IS CALLED AT SYSTEM STATE AND MUST HAVE PREVIOUSLY SAVED KISAR6.
                                ;
                                ; INPUTS:
                                ;       R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
                                ;
                                ; OUTPUTS:
                                ;       R5 = UPDATED
                                ;
                                ;-

000626                          SAVLOO:

000626  017702  000000G                 MOV     @DECPT,R2               ;; POINT TO DECNET HOME BLOCK   ; RJK03
000632  062702  000002                  ADD     #D$RNN,R2               ;; POINT TO START OF REMOTE LIST
                                ;
                                ; SCAN REMOTE TABLES FOR LOOP NODES
                                ;
000636  011246                  10$:    MOV     (R2),-(SP)              ;; SET UP UNMAPPED PTR FOR MAPPING TO RNB
000640                                  CALL    @CEACC                  ;; MAP TO RNB                     ; RJK03
000644  012602                          MOV     (SP)+,R2                ;; GET MAPPED ADDRESS
000646  001447                          BEQ     40$                     ;; IF EQ, NONE

000650  016201  000010                  MOV     R.ADD(R2),R1            ;; GET NODE ADDRESS
000654  032762  100000  000012          BIT     #RF.LOO,R.FLAG(R2)      ;; LOOP NODE ?
000662  001765                          BEQ     10$                     ;; IF EQ, NOT A LOOP NODE
000664  020164  000044                  CMP     R1,N$HLOO(R4)           ;; IS ADDRESS HIGHER THAN LOOP HIGHEST
000670  101402                          BLOS    15$                     ;; IF LOS, NO - WAS IN LIST BEFORE SO SKIP
000672  010164  000044                  MOV     R1,N$HLOO(R4)           ;; ELSE, SET LOOP HIGHEST
000676  032764  000002  000000  15$:    BIT     #NF.OVR,N$FLAG(R4)      ;; PREVIOUS OVERFLOW ?
000704  001405                          BEQ     20$                     ;; IF EQ, NO - ADD LOOP NODE
000706  032764  040000  000000          BIT     #NF.FTT,N$FLAG(R4)      ;; ELSE, IS THIS THE FIRST TIME THROUGH?
000714  001424                          BEQ     40$                     ;; IF EQ, NO - GET OUT
000716  000747                          BR      10$                     ;; ELSE, GET HIGHEST LOOP NODE ADDRESS
000720  020164  000054          20$:    CMP     R1,N$LLOO(R4)           ;; IS ADDRESS HIGHER THAN LAST LOOP HIGHEST
000724  101744                          BLOS    10$                     ;; IF LOS, NO - DON'T PUT IN LIST
000726  020164  000052                  CMP     R1,N$CLOO(R4)           ;; IS ADDRESS HIGHER THAN CURRENT LOOP HIGHEST
000732  101402                          BLOS    25$                     ;; IF LOS, NO - JUST PUT IN LIST
000734  010164  000052                  MOV     R1,N$CLOO(R4)           ;; ELSE, SET CURRENT LOOP HIGHEST
000740  052701  100000          25$:    BIS     #100000,R1              ;; MAKE IT A LOOP NODE INTERNALLY
000744  010125                          MOV     R1,(R5)+                ;; SAVE IT
000746  010446                          MOV     R4,-(SP)                ;; COPY CONTEXT AREA POINTER
000750  066416  177776                  ADD     -2(R4),(SP)             ;; POINT PAST END OF CONTEXT AREA
000754  020526                          CMP     R5,(SP)+                ;; IS THERE ENOUGH ROOM ?
000756  103727                          BLO     10$                     ;; IF LO, YES .. SAVE IT
000760  052764  000002  000000          BIS     #NF.OVR,N$FLAG(R4)      ;; ELSE, INDICATE OVERFLOW

000766                          40$:    RETURN
```

```
                                        .SBTTL  SAVADJ - SAVE ADJACENT NODE ADDRESSES
```

```
                                                                 E 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE               REFERENCES

L$TNM     000052            #5-87
L$TPT     000050            #5-87
L$TRB     000040            #5-87
L$TSZ     000070            #5-87
L$TYP     000063            #5-87
L$UNT     000003            #5-87
MAPECL  = ******  GX         12-298
MAPXPT  = ******  GX         10-203      14-385
MAXLEN  = 000006            #5-78
M$$MGE  = 000000             10-209      10-234      14-391      14-416
NF.ARE  = 000200            #5-85
NF.EPM  = 002000            #5-85
NF.EXN  = 020000            #5-85
NF.FTT  = 040000            #5-85        11-281      13-352
NF.INI  = 000001            #5-85
NF.LPB  = 000004            #5-85        6-92
NF.LV1  = 000040            #5-85        10-204      14-386
NF.LV2  = 000100            #5-85        10-204      14-386
NF.LXX  = 004000            #5-85
NF.MUX  = 000010            #5-85
NF.OVR  = 000002            #5-85        9-153       13-350      13-366      16-495      16-551      16-558      16-573      16-587
NF.PA2  = 010000            #5-85
NF.REA  = 000020            #5-85
NF.VR2  = 000400            #5-85
NF.VR3  = 001000            #5-85
NF.WLD  = 000004            #6-92        16-483
NL.ADJ  = 000010            #5-85        7-108       9-170
NL.ECL  = 000004            #5-85        7-107       9-173
NL.LOO  = 000040            #5-85        7-110       9-185
NL.REM  = 000001            #5-85        7-105       9-182
NL.ROL  = 000002            #5-85        7-106       9-176
NL.SER  = 000020            #5-85        7-109       9-179
N$ADD     000004            #5-85        16-487
N$ADJ1    000072             10-232      14-388      14-415
N$ALN   = 000002            #5-85
N$AMP     000066            #5-85        5-85
N$AMX     000056            #5-85
N$BFL     000060            #5-85       *9-166
N$BUF     000056            #5-85        5-85
N$CAR     000062            #5-85
N$CHN     000014            #5-85
N$CLOO    000052            #5-85        9-155       9-183       13-357     *13-359
N$CNO     000064            #5-85
N$ECL     000010            #5-85
N$ENC     000042             12-299
N$FLAG    000000            #5-85       *9-153       10-204      11-281      13-350      13-352     *13-366      14-386      16-483
                            *16-495     *16-551      16-558      16-573      16-587
N$HADJ    000040            #5-85        9-168       14-403     *14-405      14-420     *14-422
N$HCCX    000050            #5-85        9-154       9-167      *16-584     *16-585     *16-586
N$HECL    000036            #5-85        9-171       12-308     *12-310
N$HLCX    000046            #5-85       *9-154       10-225      10-241      11-277      12-311      14-406      14-423      15-452
N$HLOO    000044            #5-85        9-183       13-347     *13-349
```

```
                                                                 F 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
                                                    E 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-2
Symbol table

ME$NON  177777     ML$EXT  000001     MP$BBT  002165     MP$DUC  000210     MP$MDE  004421
ME$OBJ  000007     ML$FIL  000002     MP$BDF  002164     MP$DUM  000202     MP$MHO  001633
ME$OB2  000004     ML$FIR  000000     MP$BFQ  002121     MP$DUP  002127     MP$MLB  000202
ME$PRO  000005     ML$INT  000000     MP$BLK  001616     MP$DVC  002114     MP$MLK  001306
ME$SYS  000006     ML$KNO  140000     MP$BLO  001452     MP$DWE  001321     MP$MLN  001631
ME.BLO  177744     ML$MON  000003     MP$BMX  002176     MP$ELT  000157     MP$MLP  006333
ME.CON  177753     ML$SYS  000002     MP$BNP  006330     MP$ETY  001605     MP$MRB  002171
ME.CST  177765     ML$TOP  000001     MP$BRT  001620     MP$EVE  000311     MP$MRO  001605
ME.DIS  177755     MN$UNL  000377     MP$BSA  002176     MP$FNC  001752     MP$MRP  006322
ME.DON  177600     MO$ACC  000200     MP$BSD  002203     MP$GDT  002222     MP$MRT  002153
ME.FCO  177762     MO$ADD  000002     MP$BSI  002200     MP$GNM  002223     MP$MRV  006320
ME.FIO  177756     MO$ALA  000004     MP$BSP  006331     MP$GRO  002115     MP$MSB  000170
ME.FOP  177763     MO$ALI  000000     MP$BUF  000170     MP$GRP  000541     MP$MVE  000145
ME.FOR  177776     MO$CHA  000040     MP$BUP  006324     MP$GTY  002224     MP$MVI  001634
ME.FUN  177777     MO$CIR  000003     MP$BUS  001643     MP$HAD  001757     MP$MVR  001751
ME.GRO  177745     MO$CLE  000100     MP$CAC  001750     MP$HBT  002142     MP$MWN  002154
ME.HAR  177750     MO$COU  000060     MP$CAS  001762     MP$HDD  000162     MP$MXB  002172
ME.IID  177767     MO$DAC  000000     MP$CAT  002210     MP$HTM  001612     MP$MXC  000466
ME.LCO  177766     MO$DEF  000000     MP$CCS  004406     MP$HWA  002210     MP$MXR  001630
ME.LPR  177757     MO$DPR  000000     MP$CHN  002141     MP$IAT  001322     MP$MXW  002143
ME.MPR  177773     MO$ENT  000017     MP$CIR  000144     MP$IDE  000144     MP$NAA  006335
ME.MVE  177771     MO$EVE  000100     MP$CLK  002131     MP$IDP  006327     MP$NAC  001130
ME.OPE  177747     MO$INF  000160     MP$CLN  002126     MP$IHO  000215     MP$NAP  006326
ME.PLO  177751     MO$INS  000002     MP$CLT  002211     MP$INA  002177     MP$NCT  000240
ME.PMI  177743     MO$LIN  000001     MP$CMB  002142     MP$IND  002204     MP$NET  002114
ME.PNA  177752     MO$LOG  000001     MP$CMK  000537     MP$INI  002201     MP$NLI  000765
ME.PRI  177775     MO$MIR  000031     MP$CMX  002153     MP$ITI  000776     MP$NNA  000764
ME.PTY  177772     MO$NAM  000001     MP$CND  000310     MP$LAA  000231     MP$NOD  000500
ME.PVA  177760     MO$NIC  000023     MP$CNU  001753     MP$LAN  000234     MP$NRA  006334
ME.RES  177761     MO$NOD  000000     MP$COB  000311     MP$LAR  000202     MP$NRB  000214
ME.ROO  177754     MO$OFF  000001     MP$CON  002126     MP$LAS  000012     MP$NSA  001617
ME.SIZ  177774     MO$ON   000000     MP$COS  001604     MP$LBS  000203     MP$NTI  002141
ME.SYS  177746     MO$OPT  000001     MP$CPF  000230     MP$LCO  000226     MP$NUM  001642
ME.UCO  177770     MO$PER  000200     MP$CPL  000232     MP$LCT  000156     MP$NVE  001274
MF$ACT  177776     MO$PRO  000002     MP$CPT  000226     MP$LHL  000232     MP$NXN= 001476
MF$ADD  000000     MO$PRS  000001     MP$CPU  000161     MP$LLE  000227     MP$OAC  000632
MF$ADJ  177774     MO$REA  000200     MP$CSZ  001755     MP$LLO  004432     MP$OAN  000620
MF$ALL  177775     MO$SET  000000     MP$CUS  002127     MP$LMB  002152     MP$OCO  000776
MF$BYE  000300     MO$STA  000020     MP$CVA  000540     MP$LMX  002200     MP$OHO  000214
MF$CHA  000023     MO$SUM  000000     MP$DAL  002570     MP$LNA  000144     MP$ONA  000764
MF$DUM  000020     MO$TSK  000000     MP$DCO= 001464     MP$LNO  000233     MP$ONR  002114
MF$EVT  000001     MO$VOL  000000     MP$DDT  002177     MP$LOA  000170     MP$OQL  001453
MF$FUP  000264     MO$ZER  000000     MP$DEL  001131     MP$LOG  004514     MP$OTI  000777
MF$KNO  177777     MP$ACB  000012     MP$DES  000156     MP$LOO  000620     MP$OTY  001022
MF$LOA  000017     MP$ACC  000514     MP$DEV  002114     MP$LTM  001613     MP$OUS  000777
MF$LOO  177775     MP$ACT  004526     MP$DFA  001320     MP$LTY  004533     MP$OVE  001010
MF$REA  000024     MP$ADD  000766     MP$DHO= 001465     MP$LWI  000230     MP$OWN  004374
MF$SIG  177773     MP$ADJ  001440     MP$DJA  000173     MP$MAC  004420     MP$PAR  000036
MF$SPF  000302     MP$ADP  006325     MP$DLB  002571     MP$MAD  001630     MP$PAS  000513
MF$SYS  000026     MP$ADS  004406     MP$DLG  004521     MP$MAP  006323     MP$PCH  002152
MF$TES  000022     MP$ALB  000036     MP$DLI= 001466     MP$MAR  001635     MP$PCO  000024
MF$TRA  000301     MP$AMC  001640     MP$DLT  002200     MP$MAV  006321     MP$PCT  000144
MF$TRI  000021     MP$AMH  001641     MP$DRO  001441     MP$MBN  001636     MP$PDT  002114
MF$ZER  000025     MP$ANB  004420     MP$DST  000454     MP$MBR  001637     MP$PHA  000170
ML$ALL  100000     MP$ASB  000024     MP$DTE  002140     MP$MBU  001642     MP$PHY  000012
ML$CLS  000000     MP$ASC  004432     MP$DTY= 001452     MP$MCB  000156     MP$PLI  000025
ML$CON  000001     MP$AUS  000512     MP$DUA  000207     MP$MCO  001632     MP$PLN  002140
```

```
                                                    F 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-3
Symbol table
```

```
                                                        E 12
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42  Page 7-1
Symbol table

A$$CHK=  000000           LF$VR2=  000010           MB$TXO   000000           MC$P11   000001           MD$DPV   000044
A$$CPS=  000000           LF$ZER=  000002           MB$ZER   000000           MC$RFL   002046           MD$DQ    000006
A$$PRI=  000000           LP$MPT=  000010           MC$BAB   006073           MC$SBU   002051           MD$DTE   000024
A$$TRP=  000000           LP$MUX=  000004           MC$BID   001765           MC$SFL   002044           MD$DU    000002
BIT0   = 000001           LP$TRB=  000002           MC$BMC   001767           MC$SYC   000012           MD$DUP   000012
BIT1   = 000002           LP$UNT=  000001           MC$BSC   001766           MC$SYL   000036           MD$DV    000026
BIT10  = 002000           LP$WCN=  000040           MC$CAC   001442           MC$SYR   000050           MD$DZ    000030
BIT11  = 004000           LP$WDV=  000020           MC$CAP   001440           MC$SYS   000024           MD$FUL   000000
BIT12  = 010000           LP$WLD=  000360           MC$CCL   001445           MC$SYZ   000000           MD$HAL   000001
BIT13  = 020000           LP$WTR=  000200           MC$CDC   002045           MC$TBR   001140           MD$HEL   000002
BIT14  = 040000           LP$WUN=  000100           MC$CDP   001441           MC$TBS   001141           MD$KCP   000013
BIT15  = 100000           L$ADJ    000110           MC$CIF   001465           MC$TDR   002053           MD$KDP   000034
BIT2   = 000004           L$BUF    000060           MC$CLD   001464           MC$TIM   000000           MD$KDZ   000036
BIT3   = 000010           L$CHN    000054           MC$COU   100000           MC$TYP   007777           MD$KL    000040
BIT4   = 000020           L$COU    000114           MC$CTL   001454           MC$UBU   002052           MD$KMX   000054
BIT5   = 000040           L$CTB    000053           MC$CTR   001452           MC$UFD   002047           MD$KMY   000052
BIT6   = 000100           L$CTL    000002           MC$CTS   001453           MC$UMR   001132           MD$LEN   000050
BIT7   = 000200           L$DDM    000000           MC$DOV   002050           MC$UMS   001133           MD$PCL   000011
BIT8   = 000400           L$FLG    000064           MC$LBR   001750           MC$VAX   000003           MD$QNA   000005
BIT9   = 001000           L$FLX    000070           MC$LBS   001751           MC$WID   060000           MD$UNA   000001
B$10     000003           L$LEN    000124           MC$LDI   001774           MC$WIH   040000           MD$WIT   000002
B$11     000004           L$LTM    000070           MC$LDO   001775           MC$WIL   020000           MD.ABO   000017
B$12     000005           L$MSG    000056           MC$LDR   001762           MC$W08   020000           MD.ACC   000010
B$5      000000           L$MTYP   000067           MC$LDS   001763           MC$W16   040000           MD.AOB   000016
B$6      000001           L$NAM    000000           MC$LLB   002021           MC$W32   060000           MD.BOB   000011
B$7      000002           L$NLEN=  000020           MC$LLP   002115           MC$XBR   001750           MD.DIA   000007
B$7S10   000006           L$NOD    000112           MC$LLR   002007           MC$XBS   001751           MD.DOB   000015
CHKLIN   000000RG         L$NXT    000044           MC$LOO   000001           MC$XCJ   000322           MD.DUM   000002
C$$CKP=  000000           L$OPT    000062           MC$LRB   002020           MC$XCR   002260           MD.FAI   000014
C$$ORE=  000400           L$PAR    000120           MC$LRP   002114           MC$XCS   002261           MD.FNA   000001
C$$RSH=  177564           L$PDV    000055           MC$LRR   002006           MC$XDR   001762           MD.FOB   000006
D$$BUG=  177514           L$PFG    000041           MC$LSI   002032           MC$XDS   001763           MD.LOA   000001
D$$ISK=  000000           L$PLB    000116           MC$LST   002033           MC$XFR   002272           MD.LSH   000020
D$$L11=  000001           L$PRO    000067           MC$LUP   006072           MC$XFS   002273           MD.NOB   000012
D$$YNC=  000000           L$PVC    000004           MC$MAP   010000           MC$XLJ   000323           MD.NON   177777
D$$YNM=  000000           L$SCN    000020           MC$MBL   001764           MC$XLR   002330           MD.PER   000000
E$$XPR=  000000           L$SCR    000072           MC$MBX   004420           MC$XMA   000310           MD.RES   000004
F$$LVL=  000001           L$SLEN   000066           MC$MBY   001752           MC$XMC   002305           MD.ROB   000005
G$$TPP=  000000           L$SLT    000042           MC$MPX   004406           MC$XMS   002304           MD.RSH   000013
G$$TSS=  000000           L$SNM    000046           MC$NAP   001604           MC$XNR   002332           MD.SDU   000005
G$$TTK=  000000           L$SYL    000122           MC$NBR   001130           MC$XRC   002316           MD.SLO   000003
G$$WRD=  000000           L$TNM    000052           MC$NBS   001131           MC$XRR   002331           MD.SNA   000000
I$$RAR=  000000           L$TPT    000050           MC$NCR   001154           MC$XRS   002342           MD.TLO   000004
I$$RDN=  000000           L$TRB    000040           MC$NCS   001155           MC$020   000002           MD.UNA   000002
K$$CNT=  177546           L$TSZ    000070           MC$NML   001274           MD$CI    000007           MD.UNR   000003
K$$CSR=  177546           L$TYP    000063           MC$NMR   001142           MD$CNA   000003           MD.UOB   000007
K$$LDC=  000000           L$UNT    000003           MC$NMS   001143           MD$COU   000001           MD.VOL   000006
K$$TPS=  000074           L$$ASG=  000000           MC$NNO   001606           MD$DA    000010           ME$ALI   000010
LC$EXT   000000           L$$DRV=  000000           MC$NNU   001605           MD$DL    000004           ME$AL2   000003
LC$INT   000001           L$$P11=  000001           MC$NOP   001607           MD$DLV   000020           ME$ARE   000005
LD$LP  = 000000           L$$11R=  000000           MC$NOR   000000           MD$DMC   000014           ME$CIR   000003
LF$END=  000020           MB$DIS   000001           MC$NPF   001616           MD$DMF   000046           ME$EXA   000000
LF$MLT=  040000           MB$ENA   000000           MC$NPR   001630           MD$DMP   000022           ME$EXE   000200
LF$REA=  000001           MB$FUL   000002           MC$NRE   001200           MD$DMR   000050           ME$LIN   000001
LF$SEG=  100000           MB$MIX   000002           MC$NRT   001166           MD$DMV   000042           ME$LOG   000002
LF$SIG=  000040           MB$ONE   000001           MC$NVR   001642           MD$DN    000016           ME$MOD   000004
LF$SKP=  000004           MB$RXO   000001           MC$P08   000000           MD$DP    000000           ME$NOD   000000
```

```
                                                        F 12
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42  Page 7-2
Symbol table
```

```
        000020                          LP$WDV = 20             ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40             ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100            ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200            ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK

                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                ;
        000001                          LS$NTL=1                ; NTL SET FUNCTION
        000002                          LS$STA=2                ; SET STATE
        000004                          LS$COS=4                ; SET COST
        000010                          LS$OWN=10               ; SET OWNER
        000020                          LS$TAD=20               ; SET TRIBUTARY ADDRESS
        000040                          LS$ACT=40               ; SET MULTIPOINT ACTIVE
        000100                          LS$DEA=100              ; SET MULTIPOINT DEAD
        000200                          LC$NTL=200              ; NTL CLEAR FUNCTION
        000400                          LC$OWN=400              ; CLEAR OWNER
        001000                          LS$DDT=1000             ; SET DEAD TIMER
        002000                          LS$DLT=2000             ; SET DELAY TIMER
        004000                          LS$PLT=4000             ; SET POLL TIMER
        010000                          LS$BBT=10000            ; SET BABBLE TIMER
        020000                          LS$NMT=20000            ; SET NORMAL TIMER
        040000                          LS$XMT=40000            ; SET TRANSMIT TIMER
        100000                          LS$SER=100000           ; SET SERVICE [ENABLE/DISABLE]

                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FL1)
                                ;
        000001                          LS$BSA=1                ; SET ACTIVE BASE
        000002                          LS$BSD=2                ; SET DYING BASE
        000004                          LS$BSI=4                ; SET INACTIVE BASE
        000010                          LS$INA=10               ; SET ACTIVE INCREMENT
        000020                          LS$IND=20               ; SET DYING INCREMENT
        000040                          LS$INI=40               ; SET INACTIVE INCREMENT
        000100                          LS$TH1=100              ; SET DEAD THRESHOLD
        000200                          LS$TH2=200              ; SET DYING THRESHOLD
        000400                          LS$TH3=400              ; SET INACTIVE THRESHOLD
        001000                          LS$MXB=1000             ; SET MAXIMUM BLOCKS
        010000                          LS$HTM=10000            ; SET HELLO TIMER
        020000                          LS$MDT=20000            ; SET MODEM TEST
        040000                          LS$LOO=40000            ; SET CONTROLLER LOOPBACK
        100000                          LS$NOR=100000           ; SET CONTROLLER NORMAL
                                        .IF DF  R$$PRO  ; PRO/DECNET
                                ;
                                ; LOOPBACK TYPE WORD BIT DEFINITIONS (L$LTY)
                                ;
                                        LS$INT = 0              ; INTERNAL LOOPBACK
                                        LS$EXT = 1              ; EXTERNAL LOOPBACK

                                ;
                                ; MODEM TEST TYPE WORD BIT DEFINITIONS (L$MDT)
                                ;
                                        LS$LCL = 0              ; LOCAL MODEM TEST
                                        LS$REM = 1              ; REMOTE MODEM TEST
                                        LS$OFF = 2              ; SHUT OFF MODEM TESTS
                                        .ENDC   ; DF R$$PRO

                                ;
                                ; Flags word bit definitions for X.25 circuits and lines (L$FLX)
                                ;
```

```
000104  012705  000422'                 MOV     #CLECIR,R5              ; ELSE, GET CLEAR CIRCUIT STATE TABLE
                                .IFF
                                        MOV     #ME.FUN,R0              ; UNRECOGNIZED FUNCTION OR OPTION
                                        BR      30$                     ; RETURN ERROR
                                .ENDC
000110                          20$:    CALL    .NPARS                  ; PARSE THE MESSAGE
000114  103402                          BCS     30$                     ; IF CS, ERROR
000116  012700  000002                  MOV     #MS.MOR,R0              ; ELSE, SET SUCCESSFUL RETURN
                                ;
                                ; USE THE STATUS IN R0 TO BUILD THE RETURN MESSAGE AND C-BIT RETURN
                                ;
000122  016402  000060          30$:    MOV     L$BUF(R4),R2            ; RESTORE THE BUFFER ADDRESS
000126  110022                          MOVB    R0,(R2)+                ; SET THE RETURN CODE
000130  100010                          BPL     40$                     ; IF PL, SUCCESS
000132  005064  000064                  CLR     L$FLG(R4)               ; ELSE, UNSUCCESSFUL INIT
000136  016405  000072                  MOV     L$PAR(R4),R5            ; SET THE PARAMETER TYPE IN ERROR
000142  001403                          BEQ     40$                     ; IF EQ, THERE WAS NONE
000144  110522                          MOVB    R5,(R2)+                ; SET PARAMETER LOW BYTE
000146  000305                          SWAB    R5                      ; SET UP HIGH BYTE
000150  110522                          MOVB    R5,(R2)+                ; SET PARAMETER HIGH BYTE
000152  016403  000060          40$:    MOV     L$BUF(R4),R3            ; RESTORE BUFFER POINTER
000156  160302                          SUB     R3,R2                   ; CALC MESSAGE LENGTH
000160  105713                          TSTB    (R3)                    ; SUCCESS ?
000162  100001                          BPL     50$                     ; IF PL, YES
000164  000261                          SEC                             ; ELSE, NO - SET ERROR RETURN
000166                          50$:    RETURN
```

```
    910                                                 .SBTTL  DTEACT - X.25 Circuit DTE action routine
    911                                        ;+
    912                                        ; *** DTEACT - X.25 Circuit DTE action routine
    913                                        ;
    914                                        ; INPUTS
    915                                        ;       .MSGPT  - Pointer to DTE address
    916                                        ;       R4      - Address of context area
    917                                        ;
    918                                        ; OUTPUTS
    919                                        ;       Transition rejected if invalid DTE address
    920                                        ;       L$DTE(R4) - DTE address in BCD
    921                                        ;       L$DTEL(R4) - Number of digits in DTE address
    922                                        ;-
    923
    924 000726                                 DTEACT:
    925 000726                                         CALL    $SAVAL                  ; Save all registers
    926 000732  016701  000000G                        MOV     .MSGPT,R1               ; Point to DTE address
    927 000736  112102                                 MOVB    (R1)+,R2                ; Get number of digits in address
    928 000740  010264  000236                         MOV     R2,L$DTEL(R4)           ; Save length of DTE address
    929 000744  010405                                 MOV     R4,R5                   ; Point to context area
    930 000746  062705  000240                         ADD     #L$DTE,R5               ; Point to work area to get BCD address
    931 000752  012700  000010                         MOV     #<MXDTE+1>/2,R0         ; Get number of bytes required
    932 000756                                         CALL    PCKBCD                  ; Store DTE address in BCD format
    933 000762  103003                                 BCC     10$                     ; BR if valid address
    934 000764  062766  000002  000000G                ADD     #2,R$PC(SP)             ; ... else reject the transition
    935 000772                                 10$:    RETURN
```

```
                                        .SBTTL  INAACT - ACTIVE INCREMENT ACTION ROUTINE

                                ;+
                                ;
                                ; INAACT - ACTIVE INCREMENT ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 1 BYTE ACTIVE INCREMENT
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$INA(R4) - ACTIVE INCREMENT VALUE
                                ;
                                ;-
001304                          INAACT: CALL    CHKDMP                  ; CHECK FOR DMP
001310  110164  000125                  MOVB    R1,L$INA(R4)            ; SAVE ACTIVE INCREMENT VALUE
001314                                  RETURN
```

Table of contents

```
    5-   57     MACRO CALLS AND LOCAL DATA
    6-   72     ALREIN - ALIAS READ INFORMATION INITIALIZE
    7-  170     SAVALI - SAVE ALIAS CONTEXT
    8-  236     SAVCTX - SAVE AN ALIAS CONTEXT
    9-  280     FINDA - FIND A MATCHING ALIAS BLOCK
```

```
     1                                  .TITLE  ALREIN - ALIAS READ INFORMATION INITIALIZE
```

```
                                                                  F  2
ALREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:05     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE               REFERENCES

AF.INI  = 000001            #5-69       6-159
AF.OVR  = 000002            #5-69       8-268
ALREIN    000000 RG         #6-97
A$ALI     000012            #5-69       9-308       9-310       9-312
A$BUF     000006            #5-69      *6-100       6-158
A$FLG     000000            #5-69      *6-159      *6-162      *8-268
A$LEN     000024            #5-69       6-105
A$NEX     000022            #5-69      *7-202
A$NUM     000010            #5-69       6-142      *7-199      *8-275
A$OPT     000002            #5-69      *6-99        6-111       6-113
A$PAR     000004            #5-69      *6-144      *6-148      *6-153       6-163
A$SAV     000024            #5-69       7-201
A$SCO     000020            #5-69      *6-141       7-227       9-314
A$TYP     000011            #5-69       6-123       6-150       7-208       7-225
A.ACC     000020            #5-64
A.LNK     000000            #5-64
A.NAM     000002            #5-64       8-271       8-272       8-273       9-308       9-310       9-312
A.PUD     000010            #5-64
A.REM     000012            #5-64
A.UCB     000010            #5-64       7-227       8-274       9-314
CEACC   = ******   GX        7-222      9-304
CF$VR2  = ******   GX        6-146
CHKALI  = ******   GX        6-118
CHKSCO  = ******   GX        6-125
CURCTX  = ******   GX        6-145
C$FLAG  = ******   GX        6-146
DECPT   = ******   GX        7-206
D$ANN     000000             7-207
FINDA     000466 R           7-213     #9-302
KISAR6  = ******   GX        7-205     *7-233
MAXACC  = 000020            #5-69
MAXID   = 000020            #5-69
MAXPAS  = 000010            #5-69
ME$ALI    000010             6-144
ME$AL2    000003             6-148
ME.DON    177600             6-152
ME.FOR    177776             6-130
ME.FUN    177777             6-110
ME.MPR    177773             6-104
ME.UCO    177770             6-149
MF$ALL    177775             6-123      7-225
MO$CHA    000040             6-111
MS.MOR    000002             6-140
N$$VCT  = ******             7-205      7-233
SAVALI    000256 R           6-136     #7-198
SAVCTX    000406 R           7-215      7-230      #8-262
$SAVRG  = ******   GX        6-98
```

```
                                                                  G  2
ALREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:05     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00
```

```
                                                 .SBTTL  FINDA - FIND A MATCHING ALIAS BLOCK
                                       ;+
                                       ; **-FINDA-FIND A MATCHING ALIAS BLOCK
                                       ;
                                       ; THIS ROUTINE FINDS A BLOCK WITH A MATCHING ALIAS NAME AND SCOPE.
                                       ;
                                       ; INPUTS:
                                       ;        R0 = LISTHEAD ADDRESS
                                       ;        R4 = POINTER TO CONTEXT AREA WITH:
                                       ;                A$ALI = 6 CHAR ALIAS NAME
                                       ;                A$SCO = SCOPE OF ALIAS
                                       ;
                                       ; OUTPUTS:
                                       ;        IF CC:
                                       ;           R1 = MAPPED ADDRESS OF BLOCK
                                       ;        IF CS:
                                       ;           NO MATCH IN LIST
                                       ;
                                       ; REGISTERS:
                                       ;        R2, R3, R4, R5 ARE PRESERVED
                                       ;-

000640  010001                         FINDA:  MOV     R0,R1                   ; R1 IS OUTPUT REGISTER - SO COPY
000642  011146                         10$:    MOV     (R1),-(SP)              ; MOVE UNMAPPED ADDR OF NEXT BLK TO STK
000644                                         CALL    @CEACC                  ; MAP TO THE BLOCK              ; RJK02
000650  000261                                 SEC                             ; ASSUME FAILURE
000652  012601                                 MOV     (SP)+,R1                ; RECOVER MAPPED VIRTUAL ADDRESS
000654  001421                                 BEQ     40$                     ; IF EQ, THERE IS NONE (NO MATCH)
000656  026461  000012  000002                 CMP     A$ALI(R4),A.NAM(R1)     ; DOES THIS NAME MATCH ?
000664  001366                                 BNE     10$                     ; IF NE, NO
000666  026461  000014  000004                 CMP     A$ALI+2(R4),A.NAM+2(R1) ; ...
000674  001362                                 BNE     10$                     ; ...
000676  026461  000016  000006                 CMP     A$ALI+4(R4),A.NAM+4(R1) ; ...
000704  001356                                 BNE     10$                     ; ...
000706  026461  000020  000010                 CMP     A$SCO(R4),A.UCB(R1)     ; DOES THE SCOPE MATCH ?
000714  001352                                 BNE     10$                     ; IF NOT EQ, NO MATCH

000716  000241                                 CLC                             ; FOUND IT !
000720                                 40$:    RETURN

        000001                                 .END
```

Symbol table

```
AF.INI= 000001          D$NN    000060          K$$CNT= 177546          MC$MBY  001752          MC$XMC  002305
```

```
    148
    149 000142  012764  000010  000004  20$:     MOV     #ME$ALI,A$PAR(R4)          ; SET ERROR DETAIL-ASSUME NOT V2.0 NCP
    150 000150  016700  000000G                  MOV     CURCTX,R0                  ; GET ADDRESS OF CONTEXT AREA
    151 000154  032760  000000G 000000G          BIT     #CF$VR2,C$FLAG(R0)         ; CONNECTED TO V2.0 NCP?
    152 000162  001403                           BEQ     25$                        ; BR IF NO
    153 000164  012764  000003  000004           MOV     #ME$AL2,A$PAR(R4)          ; ELSE SET V2.0 ENTITY ID
    154 000172  012700  177770          25$:     MOV     #ME.UCO,R0                 ; ELSE, BAD ALIAS NAME
    155 000176  000261                           SEC
    156 000200  000401                           BR      35$
    157 000202  000241                  30$:     CLC                                ; BE SURE CARRY IS CLEAR
    158 000204                          35$:     RESRG   R2                         ; RESTORE SAVED REGISTER
    159 000206                                   RETURN
    160                                 ;
    161                                 ;  CMPCST -- compare counted strings
    162                                 ;
    163                                 ;        input:  R1 input string
    164                                 ;                R5 length of input string
    165                                 ;                R0 points to comparsion string, which is in
    166                                 ;                a byte table of the form : length, string
    167                                 ;
    168                                 ;        output: Carry clear,  match     Carry set, no match
    169                                 ;                R1, R5 saved
    170                                 ;
    171 000210                          CMPCST:
    172 000210                                   SAVRG   <R1,R5>
    173 000214  122005                           CMPB    (R0)+,R5          ; compare lengths first--must be equal
    174 000216  001403                           BEQ     10$               ; if ne, no match, but check for <6 blanks
    175 000220  121127  000040                   CMPB    (R1),#'           ; If the first char isn't blank
    176 000224  001006                           BNE     20$               ; kick it out
    177 000226  122120                  10$:     CMPB    (R1)+,(R0)+       ; Now look at characters
    178 000230  001004                           BNE     20$               ; if ne, no match
    179 000232                                   SOB     R5,10$            ; and compare next character
    180 000236  000241                           CLC                       ; Match, so ensure clear carry
    181 000240  000401                           BR      25$               ; And skip to return
    182 000242  000261                  20$:     SEC                       ; Set "no match" indicator
    183 000244                          25$:     RESRG   <R5,R1>           ; restore input string description
    184 000250                                   RETURN                    ; and get back
    185
```

```
    187                                  .SBTTL  CHKSCO - CHECK SCOPE PARAMETER
```

Table of contents

5-   49   MACRO CALLS AND LOCAL DATA
6-   68   ARREIN - READ AREA STATUS INITIALIZATION

```
      1                                          .TITLE  ARREIN - AREA READ INFORMATION INITIALIZE
```

```
ARREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:07      PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

ARRCX$           #5-54       5-65
CALL              6-131      6-142
DHBDF$           #5-55       5-57
HDRDF$           #5-55       5-58
MANDF$           #5-54       5-59
NSFDF$           #5-55       5-60
PDVDF$           #5-54       5-61
RESMAP           #5-55
RETC             #5-54       6-193
RETURN            6-195      6-217
SAVMAP           #5-55
SWSTK$            6-131
UCBDF$           #5-54       5-62
XPDDB$           #5-54       5-63
```

```
**FILE**ID**ARREOP
```

```
                                        .SBTTL  NXTNOD - STORE THE NEXT NODE
                                ;+
                                ; *** - NXTNOD - STORE THE NEXT NODE
                                ;
                                ; INPUTS:
                                        R1 = NODE ADDRESS
                                        R3 = OUTPUT BUFFER ADDRESS
                                        R4 = POINTER TO CONTEXT BLOCK
                                ;
                                ; OUTPUTS:
                                        R0-R2 = DESTROYED
                                        R3 = NEW BUFFER ADDRESS
                                ;-

001350  005701                  NXTNOD: TST     R1                              ; FOUND ONE ?
001352  001417                          BEQ     10$                             ; NO INFORMATION
001354                                  DBLOK$  PARAM,MP$NXN,CODMUL,2           ; STORE NEXT NODE ADDRESS AND NAME
001364  012700  000000'                 MOV     #TEMP,R0                        ; POINT TO BUFFER
001370  010110                          MOV     R1,(R0)                         ; COPY NEXT NODE ADDRESS
001372                                  DBLOK$  ,,DECUNS,2                      ; PUT NEXT NODE ADDRESS IN MESSAGE
001400                                  CALL    NODNAM                          ; SEARCH FOR NODE NAME
001404                                  DBLOK$  ,,ASCIMG,6                      ; STORE NODE NAME
001412  000241                  10$:    CLC
001414                                  RETURN
```

```
		.TITLE	BLNLST - BUILD NODE LIST IN CONTEXT AREA
		.IDENT	/V05.00/
		.ENABL	LC
		.NLIST	CND

;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - BUILD NODE LIST
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 4.00	13-FEB-84
;		DECNET-11M V4.0
;		DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;		DECnet-11M/S V4.2
;		DECnet-11M-Plus V3.0
;		DECnet-Micro/RSX V1.0
;
;

```

```
		SBTTL	MACRO CALLS AND LOCAL DATA
```

```
                                         .SBTTL  SAVADJ - SAVE ADJACENT NODE ADDRESSES
                                 ;+
                                 ; *** - SAVADJ - SAVE ADJACENT NODE ADDRESSES
                                 ;
                                 ; THIS ROUTINE IS CALLED AT SYSTEM STATE.
                                 ;
                                 ; INPUTS:
                                 ;       R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
                                 ;
                                 ; OUTPUTS:
                                 ;       R5 = UPDATED
                                 ;
                                 ;-

000770                           SAVADJ:
000770                                   MAPXPT                          ;; MAP AND POINT TO XPT DATABASE
000774  032764  000140  000000           BIT     #NF.LV1!NF.LV2,N$FLAG(R4) ;; ARE WE AN ENDNODE ?
001002  001447                           BEQ     30$                     ;; IF EQ, YES
001004                                   MAP     N$ADJ1(R2)              ;; GET APR BIAS FOR ADJACENCY DATA BASE
001012  001442                           BEQ     20$                     ;; IF EQ, NOT PRESENT
001014  005002                           CLR     R2                      ;; POINT TO START OF ADJACENCIES
001016                                   BIAS    R2                      ;; THRU APR 6
001026  017700  000000G                  MOV     @DECPT,R0               ;; POINT TO HOME BLOCK          ; RJK03
001032  016046  000030                   MOV     D$NLN(R0),-(SP)         ;; GET NUMBER OF LINES
001036  066016  000054                   ADD     D$NBRA(R0),(SP)         ;; ADD ROUTER ADJACENCIES
001042  066016  000056                   ADD     D$NBEA(R0),(SP)         ;; ADD ENDNODE ADJACENCIES
001046  012600                           MOV     (SP)+,R0                ;; ...
                                 ;
                                 ; SCAN THE ADJACENCY TABLE
                                 ;
001050  132762  000010  000002   10$:    BITB    #AT$UP,A$TYP(R2)        ;; IS THIS ADJACENCY UP ?
001056  001414                           BEQ     15$                     ;; IF EQ, NO - SKIP IT
001060  016201  000000                   MOV     A$NID(R2),R1            ;; GET NODE ADDRESS
001064  020164  000040                   CMP     R1,N$HADJ(R4)           ;; IS ADDRESS HIGHER THAN ADJACENCY HIGHEST
001070  101402                           BLOS    12$                     ;; IF LOS, NO - BRANCH
001072  010164  000040                   MOV     R1,N$HADJ(R4)           ;; ELSE SET HIGHEST ADJACENCY TO CURRENT ADDRESS
001076  020164  000046           12$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
001102  101402                           BLOS    15$                     ;; IF LOS, NO - RETURN
001104                                   CALL    SAVCTX                  ;; SAVE THE CONTEXT
001110  062702  000004           15$:    ADD     #A$LEN,R2               ;; GO TO NEXT NODE ENTRY
001114                                   SOB     R0,10$                  ;; IF ANOTHER NODE GET ADDRESS
001120                           20$:    RETURN
                                 ;
                                 ; ENDNODE
                                 ;
001122  016202  000072           30$:    MOV     N$ADJ1(R2),R2           ;; POINT TO ADJACENCY
001126                                   BIAS    R2                      ;; USE APR 6
001136  132762  000010  000002           BITB    #AT$UP,A$TYP(R2)        ;; IS IT UP ?
001144  001414                           BEQ     40$                     ;; IF EQ, NO
001146  016201  000000                   MOV     A$NID(R2),R1            ;; ELSE GET NODE ADDRESS
001152  020164  000040                   CMP     R1,N$HADJ(R4)           ;; IS ADDRESS HIGHER THAN ADJACENCY HIGHEST
001156  101402                           BLOS    35$                     ;; IF LOS, NO - BRANCH
001160  010164  000040                   MOV     R1,N$HADJ(R4)           ;; ELSE SET HIGHEST ADJACENCY TO CURRENT ADDRESS
001164  020164  000046           35$:    CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
001170  101402                           BLOS    40$                     ;; IF LOS, NO - RETURN
001172                                   CALL    SAVCTX                  ;; AND SAVE IT
001176                           40$:    RETURN
```

```
                                         .SBTTL  SAVSER - SAVE SERVICE NODE ADDRESS
```

```
                                                                      F 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09      PAGE 4
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

N$HREM    000032            #5-85      9-180      11-274     *11-276
N$HROL    000034            #5-85      9-174      10-222     *10-224     10-238     *10-240
N$HSER    000042            #5-85      9-177      15-449     *15-451
N$LEN     000270            #5-85
N$LFLG    000030            #5-85      9-159      *9-170     *9-173      *9-176     *9-179     *9-182     *9-185
N$LLOO    000054            #5-85     *9-155      13-355
N$LOO     000270            #5-85
N$NAM     000020            #5-85
N$NEX     000062            #5-85
N$NLEN    000017            #5-85
N$NLN   = 000200            #5-85
N$NMP     000070            #5-85      5-85
N$NMX     000060            #5-85
N$NSP     000026            #5-85      16-499     16-502
N$OPT     000002            #5-85
N$PLU     000015            #5-85
N$QCH     000016            #5-85
N$ROA1    000022             10-206    10-207
N$SAV     000064            #5-85      16-505
N$UCB     000006            #5-85
N$XPT     000012            #5-85
N$$VCT  = ******             8-131     8-133      10-206     12-300      14-388     15-447
OFFSET  = 000002            #5-80
RBNLST    000032 RG         #8-130
RF.LOO  = 100000            #5-67      11-272     13-345
R$R5    = ******  GX       *16-550    *16-561    *16-577     16-579
R$$11D  = ******             5-69
R$$11M  = 000000             5-69      10-209     10-234     14-391      14-416
R$$11S  = ******             5-69
R.ADD     000010            #5-67      11-271     13-344
R.FLAG    000012            #5-67      11-272     13-345
R.LEN     000014            #5-67
R.LNK     000000            #5-67
R.NAM     000002            #5-67
SAVADJ    000770 R           7-108    #14-384
SAVCTX    001262 R           10-227    10-243     11-279     12-313      14-408     14-425     15-454     #16-481
SAVECL    000542 R           7-107    #12-298
SAVLOO    000626 R           7-110    #13-332
SAVREM    000444 R           7-105    #11-260
SAVROL    000250 R           7-106    #10-203
SAVSER    001200 R           7-109    #15-440
SF.CIR  = 000010            #5-76
SF.DEV  = 000020            #5-76
SF.DIA  = 004000            #5-76
SF.DPA  = 000100            #5-76
SF.DPC  = 000200            #5-76
SF.DUM  = 010000            #5-76
SF.HAD  = 000004            #5-76
SF.HST  = 000002            #5-76
SF.LOA  = 002000            #5-76
SF.PH3  = 020000            #5-76
SF.PSS  = 000040            #5-76
```

```
                                                                      G 10
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09      PAGE 5
SYMBOL CROSS REFERENCE                                    CREF    04.00
```

```
                                                          F 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-3
Symbol table

MP$PLO  000012           MP$STA  000000           MS$GLO  000000           MT$NLE  000017           MX$RPS  000047
MP$PMC  004374           MP$STI  002140           MS$HOL  000002           MT$NON  000001           MX$SID  000040
MP$PNT  002126           MP$STT  002201           MS$INA  000002           MT$NR4  000005           MX$SNK  000377
MP$PRI  004411           MP$STY  000175           MS$LOA  000003           MT$NTY  000060           MX$TYP  000077
MP$PRO  002130           MP$SUB  000001           MS$LOO  000002           MT$OCT  000060           MX$UID  000020
MP$PSS  001763           MP$SUR  000156           MS$OFF  000001           MT$PHA  000002           M$$CRB= 000124
MP$PST  001762           MP$SUS  000512           MS$ON   000000           MT$POI  000000           M$$CRX= 000000
MP$RET  001631           MP$SVR  000163           MS$PRO  000006           MT$QP2  000010           M$$FCS= 000000
MP$RFA  001323           MP$SWI  002260           MS$REA  000004           MT$ROU  000000           M$$MGE= 000000
MP$RMX  002201           MP$SYP  002424           MS$REF  000001           MT$RO4  000004           M$$NET= 000000
MP$RPA  004374           MP$TFL  000156           MS$RES  000003           MT$SEC  000000           M$$OVR= 000000
MP$RPR  001606           MP$TH1  002206           MS$RST  000001           MT$SGD  000020           NM.CLN  003400
MP$RRT  002213           MP$TH2  002205           MS$RSX  000002           MT$SYS  000002           NM.INI  002400
MP$RST  002212           MP$TH3  002202           MS$RT   000005           MT$TER  000001           NM.OPR  003000
MP$RSV  001754           MP$TLN  000202           MS$SER  000002           MT$TRI  000002           NM.VR2  002401
MP$RSZ  001756           MP$TLO  000172           MS$SHU  000002           MT$TYP  007777           NM.VR3  002402
MP$RTI  001616           MP$TPA  004375           MS$SRV  000007           MT$USD  000000           N$$ACC= 000001
MP$RTM  001611           MP$TRI  002164           MS$STA  000000           MT$X25  000003           N$$BUF= 000001
MP$RVE  001604           MP$TST  000144           MS$SYN  000012           MU$INC  000001           N$$LDV= 000001
MP$RVT  000156           MP$TYP  002130           MS$TER  000001           MU$OUT  000002           N$$MCP= 000001
MP$SAC  000514           MP$UCS  004407           MS$TOP  000003           MU$PER  000000           N$$MLL= 000001
MP$SAD  000543           MP$USR  001750           MS$TRI  000005           MV$III  000000           N$$MOV= 000010
MP$SCA  000310           MP$VEC  004410           MS$UNR  000005           MV$IV   000001           N$$NCT= 000001
MP$SCO  000144           MP$VER  004406           MS$VMS  000004           MX$ACT  000020           N$$PEM= 000001
MP$SCT  002176           MP$WDF  002165           MS.DON  177600           MX$CIR  000020           PRSCIR= ****** GX
MP$SDU  000203           MP$WMX  002177           MS.MOR  000002           MX$CLN  000020           P$$P45= 000000
MP$SDV  000160           MP$XMT  002166           MS.PAR  000003           MX$CNM  000006           P$$WRD= 000000
MP$SEH  004401           MP$XPF  000242           MS.SUC  000001           MX$CON  000006           Q$$OPT= 000010
MP$SER  000144           MP$XPL  000244           MT$AR4  000003           MX$DAC  000020           R$$DER= 000000
MP$SET  000000           MP$XPT  000240           MT$ASC  000100           MX$DTE  000020           R$$K11= 000001
MP$SGZ  001644           MP$XXX  177777           MT$BIL  000001           MX$FIL  000034           R$$SND= 000000
MP$SID  000176           MP$5ST  004406           MT$BYS  000011           MX$LGC  000777           R$$11M= 000000
MP$SIN  000310           MS$ACT  000001           MT$CI   000007           MX$LIN  000020           SF$ANM  000001
MP$SLI  000156           MS$ADU  000010           MT$CLE  000077           MX$LON  000006           SF$DNM  000002
MP$SLO  000171           MS$ALO  000007           MT$COD  000200           MX$NMS  000454           S$$WRG= 000000
MP$SMX  002202           MS$ASE  000006           MT$CON  000001           MX$NOD  000006           S$$YSZ= 007600
MP$SND  000500           MS$ATR  000011           MT$DMC  000004           MX$OBJ  000006           T$$KMG= 000000
MP$SNP  006332           MS$AUT  000000           MT$ETH  000006           MX$OWN  000040           T$$MIN= 000000
MP$SNU  000542           MS$CLE  000003           MT$HEX  000040           MX$PAR  000036           V$$CTR= 001000
MP$SOB  000524           MS$DED  000004           MT$LPB  000005           MX$PAS  000010           X$$DBT= 000000
MP$SPA  000157           MS$DIE  000003           MT$MAX  000037           MX$RAC  000047           $$$SYS= 004374
MP$SPR  000536           MS$DUM  000004           MT$MUL  000100           MX$RID  000047           $$$SYX= 000000
MP$SPS  000513           MS$FAI  000013

. ABS.  000124    000   (RW,I,GBL,ABS,OVR)
        000054    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 201
Work  file writes: 135
Size of work file: 27060 Words  ( 106 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:23.05
```

```
                                                          G 11
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-4
Symbol table
```

```
                                                                F 12
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42  Page 7-2
Symbol table

ME$NON   177777          ML$EXT   000001          MP$BBT   002165          MP$DUC   000210          MP$MDE   004421
ME$OBJ   000007          ML$FIL   000002          MP$BDF   002164          MP$DUM   000202          MP$MHO   001633
ME$OB2   000004          ML$FIR   000000          MP$BFQ   002121          MP$DUP   002127          MP$MLB   000202
ME$PRO   000005          ML$INT   000000          MP$BLK   001616          MP$DVC   002114          MP$MLK   001306
ME$SYS   000006          ML$KNO   140000          MP$BLO   001452          MP$DWE   001321          MP$MLN   001631
ME.BLO   177744          ML$MON   000003          MP$BMX   002176          MP$ELT   000157          MP$MLP   006333
ME.CON   177753          ML$SYS   000002          MP$BNP   006330          MP$ETY   001605          MP$MRB   002171
ME.CST   177765          ML$TOP   000001          MP$BRT   001620          MP$EVE   000311          MP$MRO   001605
ME.DIS   177755          MN$UNL   000377          MP$BSA   002176          MP$FNC   001752          MP$MRP   006322
ME.DON   177600          MO$ACC   000200          MP$BSD   002203          MP$GDT   002222          MP$MRT   002153
ME.FCO   177762          MO$ADD   000002          MP$BSI   002200          MP$GNM   002223          MP$MRV   006320
ME.FID   177756          MO$ALA   000004          MP$BSP   006331          MP$GRO   002115          MP$MSB   000170
ME.FOP   177763          MO$ALI   000000          MP$BUF   000170          MP$GRP   000541          MP$MVE   000145
ME.FOR   177776          MO$CHA   000040          MP$BUP   006324          MP$GTY   002224          MP$MVI   001634
ME.FUN   177777          MO$CIR   000003          MP$BUS   001643          MP$HAD   001757          MP$MVR   001751
ME.GRO   177745          MO$CLE   000100          MP$CAC   001750          MP$HBT   002142          MP$MWN   002154
ME.HAR   177750          MO$COU   000060          MP$CAS   001762          MP$HDD   000162          MP$MXB   002172
ME.IID   177767          MO$DAC   000000          MP$CAT   002210          MP$HTM   001612          MP$MXC   000466
ME.LCO   177766          MO$DEF   000000          MP$CCS   004406          MP$HWA   002210          MP$MXR   001630
ME.LPR   177757          MO$DPR   000000          MP$CHN   002141          MP$IAT   001322          MP$MXW   002143
ME.MPR   177773          MO$ENT   000017          MP$CIR   000144          MP$IDE   000144          MP$NAA   006335
ME.MVE   177771          MO$EVE   000100          MP$CLK   002131          MP$IDP   006327          MP$NAC   001130
ME.OPE   177747          MO$INF   000160          MP$CLN   002126          MP$IHO   000215          MP$NAP   006326
ME.PLO   177751          MO$INS   000002          MP$CLT   002211          MP$INA   002177          MP$NCT   000240
ME.PMI   177743          MO$LIN   000001          MP$CMB   002142          MP$IND   002204          MP$NET   002114
ME.PNA   177752          MO$LOG   000001          MP$CMK   000537          MP$INI   002201          MP$NLI   000765
ME.PRI   177775          MO$MIR   000031          MP$CMX   002153          MP$ITI   000776          MP$NNA   000764
ME.PTY   177772          MO$NAM   000001          MP$CND   000310          MP$LAA   000231          MP$NOD   000500
ME.PVA   177760          MO$NIC   000023          MP$CNU   001753          MP$LAN   000234          MP$NRA   006334
ME.RES   177761          MO$NOD   000000          MP$COB   000311          MP$LAR   000202          MP$NRB   000214
ME.ROO   177754          MO$OFF   000001          MP$CON   002126          MP$LAS   000012          MP$NSA   001617
ME.SIZ   177774          MO$ON    000000          MP$COS   001604          MP$LBS   000203          MP$NTI   002141
ME.SYS   177746          MO$OPT   000001          MP$CPF   000230          MP$LCO   000226          MP$NUM   001642
ME.UCO   177770          MO$PER   000200          MP$CPL   000232          MP$LCT   000156          MP$NVE   001274
MF$ACT   177776          MO$PRO   000002          MP$CPT   000226          MP$LHL   000232          MP$NXN=  001476
MF$ADD   000000          MO$PRS   000001          MP$CPU   000161          MP$LLE   000227          MP$OAC   000632
MF$ADJ   177774          MO$REA   000200          MP$CSZ   001755          MP$LLO   004432          MP$OAN   000620
MF$ALL   177775          MO$SET   000000          MP$CUS   002127          MP$LMB   002152          MP$O[illegible]O   000776
MF$BYE   000300          MO$STA   000020          MP$CVA   000540          MP$LMX   002200          MP$O[illegible]D   000214
MF$CHA   000023          MO$SUM   000000          MP$DAL   002570          MP$LNA   000144          MP$O[illegible]A   000764
MF$DUM   000020          MO$TSK   000000          MP$DCO=  001464          MP$LNO   000233          MP$ONR   002114
MF$EVT   000001          MO$VOL   000000          MP$DDT   002177          MP$LOA   000170          MP$OQL   001453
MF$FUP   000264          MO$ZER   000000          MP$DEL   001131          MP$LOG   004514          MP$OTI   000777
MF$KNO   177777          MP$ACB   000012          MP$DES   000156          MP$LOO   000620          MP$OTY   001022
MF$LOA   000017          MP$ACC   000514          MP$DEV   002114          MP$LTM   001613          MP$OUS   000777
MF$LOO   177775          MP$ACT   004526          MP$DFA   001320          MP$LTY   004533          MP$OVE   001010
MF$REA   000024          MP$ADD   000766          MP$DHO=  001465          MP$LWI   000230          MP$OWN   004374
MF$SIG   177773          MP$ADJ   001440          MP$DIA   000173          MP$MAC   004420          MP$PAR   000036
MF$SPF   000302          MP$ADP   006325          MP$DLB   002571          MP$MAD   001630          MP$PAS   000513
MF$SYS   000026          MP$ADS   004406          MP$DLG   004521          MP$MAP   006323          MP$PCH   002152
MF$TES   000022          MP$ALB   000036          MP$DLI=  001466          MP$MAR   001635          MP$PCO   000024
MF$TRA   000301          MP$AMC   001640          MP$DLT   002200          MP$MAV   006321          MP$PCT   000144
MF$TRI   000021          MP$AMH   001641          MP$DRO   001441          MP$MBN   001636          MP$PDT   002114
MF$ZER   000025          MP$ANB   004420          MP$DST   000454          MP$MBR   001637          MP$PHA   000170
ML$ALL   100000          MP$ASB   000024          MP$DTE   002140          MP$MBU   001642          MP$PHY   000012
ML$CLS   000000          MP$ASC   004432          MP$DTY=  001452          MP$MCB   000156          MP$PLI   000025
ML$CON   000001          MP$AUS   000512          MP$DUA   000207          MP$MCO   001632          MP$PLN   002140
```

```
                                                                G 12
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42  Page 7-3
Symbol table
MP$PLO   000012          MP$SIA   000000          MS$GLO   000000          MT$NLF   000017          MX$RPS   000047
```

```
Context area

                                  ; Circuits:
          000001                           LS$CHN =      1        ; Set PVC channel
          000002                           LS$CMB =      2        ; Set maximum data
          000004                           LS$CUS =      4        ; Set circuit usage
          000010                           LS$DTE =     10        ; Set DTE
          000020                           LS$MXR =     20        ; Set maximum recalls
          000040                           LS$MXW =     40        ; Set maximum window
          000100                           LS$NUM =    100        ; Set DLM number
          000200                           LS$RET =    200        ; Set recall timer
          001000                           LS$BLK =   1000        ; Set blocking
          100000                           LP$NXC = 100000        ; Circuit is a new X.25 circuit
                                  ; Lines:
          000400                           LS$HBT =    400        ; Set holdback timer
          000002                           LS$LMB = LS$CMB        ; Set max block
          000020                           LS$MRT = LS$MXR        ; Set max retransmits
          000040                           LS$MWN = LS$MXW        ; Set max window
          000200                           LS$NTI = LS$RET        ; Set retransmit timer
          010000                           LS$PRO =  10000        ; Set line protocol
                                  ; Common:
          020000                           LS$LCT =  20000        ; Set counter timer
                                  ; Groupings:
          020053                           LS$PVC = LS$CHN!LS$CMB!LS$DTE!LS$MXW!LS$LCT
          000362                           LS$SVC = LS$CMB!LS$MXR!LS$MXW!LS$NUM!LS$RET
          001004                           LS$DLM = LS$BLK!LS$CUS
```

```
Local Symbols
```

```
    588                                                 .IF NDF R$$11S
    589
    590                                                 .SBTTL  COSACT - LINE COST ACTION ROUTINE
    591                                         ;+
    592                                         ; **-COSACT-LINE COST ACTION ROUTINE
    593                                         ;
    594                                         ; THIS ROUTINE CHECKS THAT THE LINE COST SPFCIFIED IS WITHIN RANGE,
    595                                         ; AND SAVES IT IN THE CONTEXT AREA.
    596                                         ;
    597                                         ; INPUTS:
    598                                         ;       .MSGNM  = LINE COST
    599                                         ;       R4      = POINTER TO CONTEXT AREA
    600                                         ;
    601                                         ; OUTPUTS:
    602                                         ;       IF COST IS <= 25.
    603                                         ;          L$COS(R4) = SAVED COST
    604                                         ;       ELSE, THE PARSE IS TERMINATED
    605                                         ;-
    606
    607 000170                                  COSACT:
    608 000170  116701  000000G                         MOVB    .MSGNM,R1               ; GET COST
    609 000174  003535                                  BLE     PVAERR                  ; IF LE, PARAMETER VALUE ERROR
    610 000176  020127  000031                          CMP     R1,#25.                 ; IS IT LEGAL ?
    611 000202  003132                                  BGT     PVAERR                  ; IF GT, NO - VALUE ERROR
    612 000204  110164  000101                          MOVB    R1,L$COS(R4)            ; ELSE, SAVE COST
    613 000210  000552                                  BR      PAREND                  ; END OF PARAMETER PARSE
```

```
    615                                                 .SBTTL  LCTACT - COUNTER TIMER ACTION ROUTINE
```

```
                                                  .SBTTL  NUMACT - X.25 Circuit DLM number action routine
                                          ;+
                                          ; *** NUMACT - X.25 Circuit DLM number action routine
                                          ;
                                          ; INPUTS
                                          ;         .MSGPT   - Pointer to number
                                          ;         R4       - Address of context area
                                          ;
                                          ; OUTPUTS
                                          ;         L$NUM(R4) - number in BCD
                                          ;         L$NUML(R4) - Number of digits in DTE address
                                          ;-

000774                                    NUMACT:
000774                                            CALL    $SAVAL                  ; Save all registers
001000  016701  000000G                           MOV     .MSGPT,R1               ; Point to number
001004  112102                                    MOVB    (R1)+,R2                ; Get number of digits in address
001006  010264  000236                            MOV     R2,L$NUML(R4)           ; Store number of digits in number
001012  010405                                    MOV     R4,R5                   ; Point to context area
001014  062705  000240                            ADD     #L$NUM,R5               ; Point to number in context area
001020  012700  000010                            MOV     #<MXDTE+1>/2,R0         ; Get number of bytes required
001024                                            CALL    PCKBCD                  ; Store DTE address in BCD format
001030  103003                                    BCC     10$                     ; BR if valid address
001032  062766  000002  000000G                   ADD     #2,R$PC(SP)             ; ... else reject the transition
001040                                    10$:    RETURN
```

```
                                            .SBTTL  INDACT - DYING INCREMENT ACTION ROUTINE

                                    ;+
                                    ;
                                    ; INDACT - DYING INCREMENT ACTION ROUTINE
                                    ;
                                    ; INPUTS:
                                    ;       .MSGPT - POINTER TO 1 BYTE DYING INCREMENT
                                    ;       R4     - POINTER TO CONTEXT AREA
                                    ;
                                    ; OUTPUTS:
                                    ;       L$IND(R4) - DYING INCREMENT VALUE
                                    ;
                                    ;-
001316                              INDACT: CALL    CHKDMP                  ; CHECK FOR DMP
001322  110164  000126                      MOVB    R1,L$IND(R4)            ; SAVE DYING INCREMENT VALUE
001326                                      RETURN
```

```
                                            .SBTTL  INIACT - INACTIVE INCREMENT ACTION ROUTINE
```

```
          .TITLE  ALREIN - ALIAS READ INFORMATION INITIALIZE
          .IDENT  /V05.00/
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - ALIAS READ INFORMATION INITIALIZE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;


;
; 4.03  17-OCT-84               RJK
;       Change Symbols for Vectoring Comm Exec
;
```

```
          .SBTTL  MACRO CALLS AND LOCAL DATA
```

G 2

```
ALREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:05     PAGE 2
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

ALRCX$           #5-62        5-69
ANBDF$           #5-61        5-64
CALL              6-118       6-125       6-136       7-204       7-213       7-215       7-222       7-230       9-304
DHBDF$           #5-62        5-67
HDRDF$           #5-62        5-66
MANDF$           #5-61        5-65
MAP              #5-61
RESMAP           #5-61        7-233
RETURN            6-168       7-234       8-277       9-318
SAVMAP           #5-61        7-205
SWSTK$           #5-61        7-204
```

H 2

```
**FILE**ID**ALREOP
```

```
                                                                 G 3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-1
Symbol table

AF.INI= 000001           D$NN    000060           K$$CNT= 177546           MC$MBY  001752           MC$XMC  002305
AF.OVR= 000002           D$OUTT  000043           K$$CSR= 177546           MC$MPX  004406           MC$XMS  002304
ALREOP  000060RG         D$RETF  000050           K$$LDC= 000000           MC$NAP  001604           MC$XNR  002332
A$[illegible]I  000012         D$RNN   000002           K$$TPS= 000074           MC$NBR  001130           MC$XRC  002316
A$BUF   000006           D$RTMR  000076           LC$EXT  000000           MC$NBS  001131           MC$XRR  002331
A$FLG   000000           D$SEG   000036           LC$INT  000001           MC$NCR  001154           MC$XRS  002342
A$LEN   000024           D$SER   000032           LD$LP = 000000           MC$NCS  001155           MC$020  000002
A$NEX   000022           D$SQRL  000052           L$$ASG= 000000           MC$NML  001274           MD$CI   000007
A$NUM   000010           D$$BUG= 177514           L$$DRV= 000000           MC$NMR  001142           MD$CNA  000003
A$OPT   000002           D$$ISK= 000000           L$$P11= 000001           MC$NMS  001143           MD$COU  000001
A$PAR   000004           D$$L11= 000001           L$$11R= 000000           MC$NNO  001606           MD$DA   000010
A$SAV   000024           D$$YNC= 000000           MAXACC= 000020           MC$NNU  001605           MD$DL   000004
A$SCO   000020           D$$YNM= 000000           MAXID = 000020           MC$NOP  001607           MD$DLV  000020
A$TYP   000011           E$$XPR= 000000           MAXPAS= 000010           MC$NOR  000000           ML$DMC  000014
A$$CHK= 000000           FINDA   000640R          MB$DIS  000001           MC$NPF  001616           MD$DMF  000046
A$$CPS= 000000           FMTDES  000446R          MB$ENA  000000           MC$NPR  001630           MD$DMP  000022
A$$PRI= 000000           FMTNAM  000306R          MB$FUL  000002           MC$NRE  001200           MD$DMR  000050
A$$TRP= 000000           FMTSCO  000354R          MB$MIX  000002           MC$NRT  001166           MD$DMV  000042
A.ACC   000020           F$$LVL= 000001           MB$ONE  000001           MC$NVR  001642           MD$DN   000016
A.LNK   000000           G$$TPP= 000000           MB$RXO  000001           MC$P08  000000           MD$DP   000000
A.NAM   000002           G$$TSS= 000000           MB$TXO  000000           MC$P11  000001           MD$DPV  000044
A.PUD   000010           G$$TTK= 000000           MB$ZER  000000           MC$RFL  002046           MD$DQ   000006
A.REM   000012           G$$WRD= 000000           MC$BAB  006073           MC$SBU  002051           MD$DTE  000024
A.UCB   000010           H.CSP   000000           MC$BID  001765           MC$SFL  002044           MD$DU   000002
CEACC = ****** GX        H.CUIC  000010           MC$BMC  001767           MC$SYC  000012           MD$DUP  000012
C$$CKP= 000000           H.DSW   000046           MC$BSC  001766           MC$SYL  000036           MD$DV   000026
C$$ORE= 000400           H.DUIC  000012           MC$CAC  001442           MC$SYR  000050           MD$DZ   000030
C$$RSH= 177564           H.EFLM  000004           MC$CAP  001440           MC$SYS  000024           MD$FUL  000000
DECPT = ****** GX        H.EFSV  000040           MC$CCL  001445           MC$SYZ  000000           MD$HAL  000001
D$AMXC  000072           H.FCS   000050           MC$CDC  002045           MC$TBR  001140           MD$HEL  000002
D$AMXH  000074           H.FORT  000052           MC$CDP  001441           MC$TBS  001141           MD$KCP  000013
D$ANN   000000           H.FPSA  000042           MC$CIF  001465           MC$TDR  002053           MD$KDP  000034
D$BRPR  000102           H.FPSL= 000062           MC$CLD  001464           MC$TIM  000000           MD$KDZ  000036
D$BRTM  000100           H.FPVA  000034           MC$COU  100000           MC$TYP  007777           MD$KL   000040
D$DELF  000045           H.GARD  000072           MC$CTL  001454           MC$UBU  002052           MD$KMX  000054
D$DELW  000046           H.HDLN  000002           MC$CTR  001452           MC$UFD  002047           MD$KMY  000052
D$END = 000104           H.IPC   000016           MC$CTS  001453           MC$UMR  001132           MD$LEN  000050
D$FNB   000034           H.IPS   000014           MC$DOV  002050           MC$UMS  001133           MD$PCL  000011
D$HIOR  000024           H.ISP   000020           MC$LBR  001750           MC$VAX  000003           [illegible]NA  000005
D$HOST  000022           H.LUN   000076           MC$LBS  001751           MC$WID  060000           [illegible]$UNA  000001
D$INAC  000044           H.NLUN  000074           MC$LDI  001774           MC$WIH  040000           MD$WIT  000002
D$INCT  000042           H.NML   000061           MC$LDO  001775           MC$WIL  020000           MD.ABO  000017
D$IPL   000051           H.ODVA  000022           MC$LDR  001762           MC$W08  020000           MD.ACC  000010
D$LID   000020           H.ODVL  000024           MC$LDS  001763           MC$W16  040000           MD.AOB  000016
D$LNAM  000006           H.OVLY  000054           MC$LLB  002021           MC$W32  060000           MD.BOB  000011
D$LNUM  000014           H.PFVA  000032           MC$LLP  002115           MC$XBR  001750           MD.DIA  000007
D$LST   000047           H.RCVA  000036           MC$LLR  002007           MC$XBS  001751           MD.DOB  000015
D$MAXC  000064           H.RRVA  000062           MC$LOO  000001           MC$XCJ  000322           MD.DUM  000002
D$MAXH  000066           H.SPRI  000060           MC$LRB  002020           MC$XCR  002260           MD.FAI  000014
D$MAXV  000070           H.TKVA  000026           MC$LRP  002114           MC$XCS  002261           MD.FNA  000001
D$MLL   000040           H.TKVL  000030           MC$LRR  002006           MC$XDR  001762           MD.FOB  000006
D$MNOD  000041           H.VEXT  000056           MC$LSI  002032           MC$XDS  001763           MD.LOA  000001
D$NA    000062           H.WND   000044           MC$LST  002033           MC$XFR  002272           MD.LSH  000020
D$NBEA  000056           H.X25   000064           MC$LUP  006072           MC$XFS  002273           MD.NOB  000012
D$NBRA  000054           I$$RAR= 000000           MC$MAP  010000           MC$XLJ  000323           MD.NON  177777
D$NEND= 000054           I$$RDN= 000000           MC$MBL  001764           MC$XLR  002330           MD.PER  000000
D$NLN   000030           KISAR6= ****** GX        MC$MBX  004420           MC$XMA  000310           MD.RES  000004
```

```
                                                                 H 3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-2
Symbol table
```

```
                                        .SBTTL  CHKSCO - CHECK SCOPE PARAMETER
                                ;+
                                ; **-CHKSCO-CHECK SCOPE PARAMETER
                                ;
                                ; THIS ROUTINE CHECKS THAT THE FIRST PARAMETER IS THE "SCOPE"
                                ; OF THE OPERATION, AND IF VALID, FORMATS AND SAVES THE SCOPE IN
                                ; THE CONTEXT AREA.
                                ;
                                ; INPUTS:
                                ;       R2 = LENGTH OF DATA IN INPUT BUFFER
                                ;       R3 = ADDRESS OF INPUT BUFFER
                                ;       R4 = ADDRESS OF CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       R2 = LENGTH OF REMAINING DATA INPUT BUFFER
                                ;       R3 = POINTER TO REMAINING DATA IN INPUT BUFFER
                                ;       C-BIT ERROR INDICATOR
                                ;         IF CLEAR, A$ACO(R4) = SAVED SCOPE, AS 0 (GLOBAL) OR TERMINAL UCB
                                ;         IF SET, ERROR RETURN IN R0, SCOPE PARAMETER SET IN A$PAR
                                ;
                                ; REGISTERS:
                                ;       R4 IS PRESERVED
                                ;-

000252                          CHKSCO::
                                ;
                                ; CHECK PARAMETER TYPE
                                ;
000252  012701  000144                  MOV     #MP$SCO,R1              ; SET PARAMETER TYPE TO CHECK FOR
000256  010164  000004                  MOV     R1,A$PAR(R4)            ; CURRENT PARAMETER = SCOPE
000262                                  CALL    CHKPAR                  ; CHECK FOR THE SCOPE PARAMETER
000266  103516                          BCS     40$                     ; IF CS, ERROR - RETURN
                                ;
                                ; CHECK SCOPE TYPE
                                ;
000270  005064  000020                  CLR     A$SCO(R4)               ; ASSUME SCOPE IS GLOBAL
000274  005302                          DEC     R2                      ; SUB SCOPE TYPE FROM BUFFER COUNT

                                        .IF NDF R$$PRO  ; PRO/DECNET

000276  002512                          BLT     40$                     ; IF LT, SCOPE WAS GLOBAL

                                        .IFF    ; NDF R$$PRO

                                        BLT     30$                     ; IF LT, SCOPE WAS GLOBAL-ERROR

                                        .ENDC   ; NDF R$$PRO

000300  112305                          MOVB    (R3)+,R5                ; ELSE, GET TYPE INDICATOR

                                        .IF NDF R$$PRO  ; PRO/DECNET

000302  001510                          BEQ     40$                     ; IF EQ, GLOBAL

                                        .IFF    ; NDF R$$PRO

                                        BEQ     30$                     ; IF EQ, GLOBAL - ERROR
```

```
                .TITLE  ARREIN - AREA READ INFORMATION INITIALIZE
                .IDENT  /V05.00/
                .ENABL  LC

;
; COPYRIGHT (C) 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - AREA READ INFORMATION INITIALIZE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  25-MAR-83       Dave Rachlis
;       INITIAL CREATION.
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
**FILE**ID**ARREOP
```

```
  AAAAAA     RRRRRRRR     RRRRRRRR     EEEEEEEEEE     OOOOOO     PPPPPPPP
  AAAAAA     RRRRRRRR     RRRRRRRR     EEEEEEEEEE     OOOOOO     PPPPPPPP
AA      AA   RR      RR   RR      RR   EE             OO      OO   PP      PP
AA      AA   RR      RR   RR      RR   EE             OO      OO   PP      PP
AA      AA   RR      RR   RR      RR   EE             OO      OO   PP      PP
AA      AA   RR      RR   RR      RR   EE             OO      OO   PP      PP
AA      AA   RRRRRRRR     RRRRRRRR     EEEEEEEE       OO      OO   PPPPPPPP
AA      AA   RRRRRRRR     RRRRRRRR     EEEEEEEE       OO      OO   PPPPPPPP
AAAAAAAAAA   RR  RR       RR  RR       EE             OO      OO   PP
AAAAAAAAAA   RR  RR       RR  RR       EE             OO      OO   PP
AA      AA   RR    RR     RR    RR     EE             OO      OO   PP                ....
AA      AA   RR    RR     RR    RR     EE             OO      OO   PP                ....
AA      AA   RR      RR   RR      RR   EEEEEEEEEE     OOOOOO     PP                ....
AA      AA   RR      RR   RR      RR   EEEEEEEEEE     OOOOOO     PP                ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07
Table of contents
```

```
    425
    426                                                  .SBTTL  NODNAM - FIND AND FORMAT NODE NAME
    427                                         ;+
    428                                         ; *** - NODNAM - FIND AND FORMAT NODE NAME
    429                                         ;
    430                                         ; INPUTS:
    431                                         ;        R1 = NODE ADDRESS OR CHANNEL #
    432                                         ;        R3 = OUTPUT BUFFER ADDRESS
    433                                         ;
    434                                         ; OUTPUTS:
    435                                         ;        R3 = NEXT BYTE IN OUTPUT BUFFER
    436                                         ;        R0 = DESTROYED
    437                                         ;
    438                                         ;-
    439 001416                                  NODNAM:
    440 001416                                          SAVRG   <R1,R2>                 ; SAVE REGISTERS
    441 001422  010400                                  MOV     R4,R0                   ; POINT TO NODE NAME BUFFER
    442 001424  062700  000020                          ADD     #A$NAM,R0               ; ...
    443 001430  017702  000000G                         MOV     @DECPT,R2               ; POINT TO DECNET HOMEBLOCK     ; RJK03
    444 001434  026201  000014                          CMP     D$LNUM(R2),R1           ; REMOTE NODE ?
    445 001440  001014                                  BNE     5$                      ; IF NE, YES
    446 001442  062702  000006                          ADD     #D$LNAM,R2              ; POINT TO EXECUTOR NODE NAME
    447         000006                                  .REPT   6
    448                                                 MOVB    (R2)+,(R0)+             ; MOVE IT INTO SCRATCH BUFFER
    449                                                 .ENDR
    450 001462  162700  000006                          SUB     #6,R0                   ; POINT TO START OF STRING
    451 001466  010002                                  MOV     R0,R2                   ; AGAIN
    452 001470  000404                                  BR      7$                      ; JOIN COMMON CODE
    453
    454 001472  010002                          5$:     MOV     R0,R2                   ; COPY POINTER TO BUFFER
    455 001474                                          CALL    MAPADD                  ; TRY TO FIND NODE NAME
    456 001500  103406                                  BCS     20$                     ; IF CS, NO NODE NAME
    457
    458 001502  062702  000006                  7$:     ADD     #6,R2                   ; POINT TO END OF NODE NAME
    459 001506  124227  000040                  10$:    CMPB    -(R2),#'                ; REMOVE SPACES AT END OF NAME
    460 001512  001775                                  BEQ     10$                     ; ...
    461 001514  005202                                  INC     R2                      ; POINT PAST END OF NAME
    462 001516  160002                          20$:    SUB     R0,R2                   ; CALCULATE LENGTH OF NAME
    463 001520  110240                                  MOVB    R2,-(R0)                ; STORE NODE NAME LENGTH
    464 001522                                          RESRG   <R2,R1>                 ; RESTORE R2
    465 001526                                          RETURN
    466
    467         000001                                  .END
```

```
     47                                            .SBTTL  MACRO CALLS AND LOCAL DATA
     48                                    ;
     49                                    ; LIBRARY MACRO CALLS
     50                                    ;
     51                                            .MCALL  MANDF$,RNBDF$,SAVMAP,RESMAP,MAP,SAVRG,RESRG,ADJDF$
     52                                            .MCALL  ECDDB$,CTRDF$,PDVDF$,BIAS,ALUN$S,HDRDF$,XPTDF$,XPDDB$
     53                                            .MCALL  IMSG$,DHBDF$,NORCX$,CIRCX$,SERDF$,CEACC$
     54
     55                                    ;
     56                                    ; LOCAL MACROS
     57                                    ;
     58                                            .MACRO  MAPECL
     59                                            JSR     R2,MAPECL
     60                                            .ENDM
     61
     62                                            .MACRO  MAPXPT
     63                                            JSR     R2,MAPXPT
     64                                            .ENDM
     65
     66 000000                                     MANDF$                          ; DEFINE NETWORK MANAGEMENT FLAGS
     67 000000                                     RNBDF$                          ; DEFINE REMOTE BLOCK FORMAT
     68 000000                                     CTRDF$                          ; DEFINE COUNTER TABLES
     69 000000                                     PDVDF$                          ; DEFINE PDV OFFSETS
     70 000000                                     ECDDB$                          ; DEFINE ECL DATABASE OFFSETS
     71 000000                                     HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     72 000000                                     XPDDB$                          ; DEFINE XPT DATABASE OFFSETS
     73 000000                                     XPTDF$                          ; DEFINE XPT OFFSETS
     74 000000                                     ADJDF$                          ; DEFINE ADJACENCY OFFSETS
     75 000000                                     DHBDF$                          ; DEFINE DEC HOME BLOCK
     76 000000                                     SERDF$                          ; DEFINE SERVICE BLOCK OFFSETS
     77
     78               000006                       MAXLEN = 6                      ; MAXIMUM LENGTH OF NODE NAME
     79
     80               000002                       OFFSET = 2                      ; OFFSET TO SAVED REGISTERS
     81
     82                                    ;
     83                                    ; CONTEXT AREA OFFSET DEFINITIONS
     84                                    ;
     85 000000                                     NORCX$  LIST
                                           ;
                                           ; CONTEXT AREA OFFSET DEFINITIONS
                                           ;
        000052                                     .ASECT
                      000000                       .=0
        000000                             N$FLAG: .BLKW   1                       ; FLAGS WORD
        000002                             N$OPT:  .BLKW   1                       ; OPTIONS WORD
        000004                             N$ADD:  .BLKW   1                       ; NODE ADDRESS OR TYPE
        000006                             N$UCB:  .BLKW   1                       ; NS: UCB ADDRESS
        000010                             N$ECL:  .BLKW   1                       ; ECL'S PDV ADDRESS
        000012                             N$XPT:  .BLKW   1                       ; XPT'S PDV ADDRESS
        000014                             N$CHN:  .BLKB   1                       ; CHANNEL NUMBER
        000015                             N$PLU:  .BLKB   1                       ; PLURAL INFO TYPE
        000016                             N$QCH:  .BLKB   1                       ; QUALIFIER CHANNEL NUMBER
        000017                             N$NLEN: .BLKB   1                       ; NODE NAME LENGTH (ASCII IMAGE)
        000020                             N$NAM:  .BLKW   3                       ; NODE NAME BUFFER
        000026                             N$NSP:  .BLKW   1                       ; FOR EXEC ADDRESS
        000030                             N$LFLG: .BLKW   1                       ; LIST FLAG WORD
```

```
                                        .SBTTL  SAVSER - SAVE SERVICE NODE ADDRESS
                               ;+
                               ; ** SAVSER - SAVE SERVICE NODE ADDRESSES
                               ;
                               ; INPUTS:        NONE.
                               ;
                               ; OUTPUT:        ALL NODE ADDRESSES IN THER SERVICE
                               ;                DATABASE ARE SAVED IN THE CONTEXT AREA.
                               ;
                               ; REGISTERS:     R0 AND R1 ARE DESTROYED.
                               ;-

001200                         SAVSER:
001200  017700  000000G                 MOV     @DECPT,R0               ;; POINT TO DECNET HOMEBLOCK    ; RJK03
001204  001425                          BEQ     20$                     ;; IF EQ, NOT SET UP
001206  062700  000032                  ADD     #D$SER,R0               ;; POINT TO ADDRESS OF FIRST BLOCK

001212  011000                 10$:     MOV     (R0),R0                 ;; GET UNMAPPED ADDRESS
001214  001421                          BEQ     20$                     ;; IF EQ, END OF LIST
001216                                  CEACC$  R0                      ;; MAP IT
001226  016001  000004                  MOV     S.ADD(R0),R1            ;; PICK UP NODE ADDRESS
001232  020164  000042                  CMP     R1,N$HSER(R4)           ;; IS ADDRESS HIGHER THAN SERVICE HIGHEST
001236  101402                          BLOS    15$                     ;; IF LOS, NO - BRANCH
001240  010164  000042                  MOV     R1,N$HSER(R4)           ;; ELSE SET HIGHEST SERVICE TO CURRENT ADDRESS
001244  020164  000046         15$:     CMP     R1,N$HLCX(R4)           ;; IS ADDRESS HIGHER THAN LAST CONTEXT HIGHEST
001250  101760                          BLOS    10$                     ;; IF LOS, NO - RETURN
001252                                  CALL    SAVCTX                  ;; SAVE IT
001256  000755                          BR      10$                     ;; CONTINUE

001260                         20$:     RETURN
```

```
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09     PAGE 5
SYMBOL  CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE              REFERENCES

SF.SEC   = 000400           #5-76
SF.TER   = 001000           #5-76
S.ADD      000004           #5-76         15-448
S.CIR      000016           #5-76
S.DEV      000020           #5-76
S.DIA      000046           #5-76
S.DPA      000032           #5-76
S.DPC      000036           #5-76
S.DUM      000050           #5-76
S.FLAG     000002           #5-76
S.HAD      000010           #5-76
S.HST      000006           #5-76
S.LENG     000052           #5-76
S.LNK      000000           #5-76
S.LOA      000044           #5-76
S.PSS      000022           #5-76
S.SEC      000040           #5-76
S.TER      000042           #5-76
T$FLAG     000044           #5-68
T$LIF      000013           #5-68
T$LIFL     000013           #5-68
T$LIFO     000013           #5-68
T$LIFS     000013           #5-68
T$LIN      000000           #5-68
T$LIPS     000006           #5-68
T$LLD      000012           #5-68
T$LLDC     000045           #5-68
T$LLDL     000012           #5-68
T$LLDO     000012           #5-68
T$LLDS     000012           #5-68
T$LLEN     000046           #5-68
T$LOPR     000002           #5-68
T$LTCL     000024           #5-68
T$LTIM     000026           #5-68
T$LTPR     000014           #5-68
T$LTPS     000020           #5-68
T$NAPL     000004           #5-68
T$NFE      000000           #5-68
T$NLEN     000010           #5-68
T$NNUL     000002           #5-68
T$NOPL     000006           #5-68
T$NRNI     000042           #5-68
T$NRPL     000005           #5-68
T$NRUL     000007           #5-68
T$NVR      000001           #5-68
T$RPRI     000040           #5-68
T$SVC      000034           #5-68
T$T5       000030           #5-68
T$T6       000032           #5-68
X$$MCB   = ******            5-69         5-69
ZF.COU   = 001000           #5-69
ZF.DDM   = 000001           #5-69
```

```
BLNLST      CREATED BY  MACRO  ON 29-JUN-85 AT 12:09     PAGE 6
SYMBOL  CROSS REFERENCE                                  CREF   04.00
```

G 11

```
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41  Page 7-4
Symbol table

SY:CHKCIR.V2,[135,134]CHKCIR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CHKCIR
```

H 11

```
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00
```

```
MP$PLO   000012      MP$STA   000000      MS$GLO   000000      MT$NLE   000017      MX$RPS   000047
MP$PMC   004374      MP$STI   002140      MS$HOL   000002      MT$NON   000001      MX$SID   000040
MP$PNT   002126      MP$STT   002201      MS$INA   000002      MT$NR4   000005      MX$SNK   000377
MP$PRI   004411      MP$STY   000175      MS$LOA   000003      MT$NTY   000060      MX$TYP   000077
MP$PRO   002130      MP$SUB   000001      MS$LOO   000002      MT$OCT   000060      MX$UID   000020
MP$PSS   001763      MP$SUR   000156      MS$OFF   000001      MT$PHA   000002      M$$CRB=  000124
MP$PST   001762      MP$SUS   000512      MS$ON    000000      MT$POI   000000      M$$CRX=  000000
MP$RET   001631      MP$SVR   000163      MS$PRO   000006      MT$QP2   000010      M$$FCS=  000000
MP$RFA   001323      MP$SWI   002260      MS$REA   000004      MT$ROU   000000      M$$MGE=  000000
MP$RMX   002201      MP$SYP   002424      MS$REF   000001      MT$R04   000004      M$$NET=  000000
MP$RPA   004374      MP$TFL   000156      MS$RES   000003      MT$SEC   000000      M$$OVR=  000000
MP$RPR   001606      MP$TH1   002206      MS$RST   000001      MT$SGD   000020      NM.CLN   003400
MP$RRT   002213      MP$TH2   002205      MS$RSX   000002      MT$SYS   000002      NM.INI   002400
MP$RST   002212      MP$TH3   002202      MS$RT    000005      MT$TER   000001      NM.OPR   003000
MP$RSV   001754      MP$TLN   000202      MS$SER   000002      MT$TRI   000002      NM.VR2   002401
MP$RSZ   001756      MP$TLO   000172      MS$SHU   000002      MT$TYP   007777      NM.VR3   002402
MP$RTI   001616      MP$TPA   004375      MS$SRV   000007      MT$USD   000000      N$$ACC=  000001
MP$RTM   001611      MP$TRI   002164      MS$STA   000000      MT$X25   000003      N$$BUF=  000001
MP$RVE   001604      MP$TST   000144      MS$SYN   000012      MU$INC   000001      N$$LDV=  000001
MP$RVT   000156      MP$TYP   002130      MS$TER   000001      MU$OUT   000002      N$$MCP=  000001
MP$SAC   000514      MP$UCS   004407      MS$TOP   000003      MU$PER   000000      N$$MLL=  000001
MP$SAD   000543      MP$USR   001750      MS$TRI   000005      MV$III   000000      N$$MOV=  000010
MP$SCA   000310      MP$VEC   004410      MS$UNR   000005      MV$IV    000001      N$$NCT=  000001
MP$SCO   000144      MP$VER   004406      MS$VMS   000004      MX$ACT   000020      N$$PEM=  000001
MP$SCT   002176      MP$WDF   002165      MS.DON   177600      MX$CIR   000020      PRSLIN=  ****** GX
MP$SDU   000203      MP$WMX   002177      MS.MOR   000002      MX$CLN   000020      P$$P45=  000000
MP$SDV   000160      MP$XMT   002166      MS.PAR   000003      MX$CNM   000006      P$$WRD=  000000
MP$SEH   004401      MP$XPF   000242      MS.SUC   000001      MX$CON   000006      Q$$OPT=  000010
MP$SER   000144      MP$XPL   000244      MT$AR4   000003      MX$DAC   000020      R$$DER=  000000
MP$SET   000000      MP$XPT   000240      MT$ASC   000100      MX$DTE   000020      R$$K11=  000001
MP$SGZ   001644      MP$XXX   177777      MT$BIL   000001      MX$FIL   000034      R$$SND=  000000
MP$SID   000176      MP$5ST   004406      MT$BYS   000011      MX$LGC   000777      R$$11M=  000000
MP$SIN   000310      MS$ACT   000001      MT$CI    000007      MX$LIN   000020      SF$ANM   000001
MP$SLI   000156      MS$ADU   000010      MT$CLE   000077      MX$LON   000006      SF$DNM   000002
MP$SLO   000171      MS$ALO   000007      MT$COD   000200      MX$NMS   000454      S$$WRG=  000000
MP$SMX   002202      MS$ASE   000006      MT$CON   000001      MX$NOD   000006      S$$YSZ=  007600
MP$SND   000500      MS$ATR   000011      MT$DMC   000004      MX$OBJ   000006      T$$KMG=  000000
MP$SNP   006332      MS$AUT   000000      MT$ETH   000006      MX$OWN   000040      T$$MIN=  000000
MP$SNU   000542      MS$CLE   000003      MT$HEX   000040      MX$PAR   000036      V$$CTR=  001000
MP$SOB   000524      MS$DED   000004      MT$LPB   000005      MX$PAS   000010      X$$DBT=  000000
MP$SPA   000157      MS$DIE   000003      MT$MAX   000037      MX$RAC   000047      $$$SYS=  004374
MP$SPR   000536      MS$DUM   000004      MT$MUL   000100      MX$RID   000047      $$$SYX=  000000
MP$SPS   000513      MS$FAI   000013

. ABS.   000124    000   (RW,I,GBL,ABS,OVR)
         000054    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 201
Work  file writes: 135
Size of work file: 27040 Words  ( 106 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:22.55
```

```
     74                                          .SBTTL  Local Symbols
     75
     76          007777                          MAXCHN = 4095.                ; Maximum PVC channel number
     77          000017                          MXDTE  = 15.                  ; Max length of DTE address
     78
```

```
                                                  .SBTTL  LCTACT - COUNTER TIMER ACTION ROUTINE

                                          ;+
                                          ; LCTACT - SET COUNTER TIMER ACTION ROUTINE
                                          ;
                                          ; THIS ROUTINE STORES THE SPECIFIED COUNTER TIMER
                                          ;
                                          ; INPUTS:
                                          ;        .MSGPT - POINTER TO COUNTER TIMER
                                          ;        R4 - ADDRESS OF CONTEXT AREA
                                          ;
                                          ; OUTPUTS:
                                          ;        L$LCT(R4) - COUNTER TIMER VALUE
                                          ;
                                          ;-
000212  016701  000000G                   LCTACT: MOV     .MSGPT,R1               ; POINT TO COUNTER TIMER VALUE
000216  112164  000224                            MOVB    (R1)+,L$LCT(R4)         ; STORE LOW BYTE OF COUNTER TIMER
000222  111164  000225                            MOVB    (R1),L$LCT+1(R4)        ; STORE COUNTER TIMER VALUE
000226                                            RETURN
```

```
    963                                                 .SBTTL  MXRACT - X.25 Circuit maximum recalls action routine
    964                                         ;+
    965                                         ; *** MXRACT - X.25 Circuit maximum recalls action routine
    966                                         ;
    967                                         ; INPUTS
    968                                         ;       .MSGPT  - Pointer to maximum recalls parameter value
    969                                         ;       R4      - Pointer to context area
    970                                         ;
    971                                         ; OUTPUTS
    972                                         ;       L$MXR(R4) Contains maximum recalls parameter
    973                                         ;-
    974
    975 001042                                  MXRACT:
    976 001042  016701  000000G                         MOV     .MSGPT,R1               ; Get pointer to data
    977 001046  112164  000253                          MOVB    (R1)+,L$MXR(R4)         ; Get data
    978 001052                                          RETURN
```

```
                                        .SBTTL  INIACT - INACTIVE INCREMENT ACTION ROUTINE

                                ;+
                                ;
                                ; INIACT - INACTIVE INCREMENT ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 1 BYTE INACTIVE INCREMENT
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$INI(R4) - INACTIVE INCREMENT VALUE
                                ;
                                ;-
001330                          INIACT: CALL    CHKDMP                  ; CHECK FOR DMP
001334  110164  000127                  MOVB    R1,L$INI(R4)            ; SAVE INACTIVE INCREMENT VALUE
001340                                  RETURN
```

```
                                    .SBTTL  MACRO CALLS AND LOCAL DATA
                               ;
                               ; MACRO LIBRARY CALLS
                               ;
                                    .MCALL  ANBDF$,MANDF$,SWSTK$,SAVMAP,MAP,RESMAP
                                    .MCALL  HDRDF$,ALRCX$,DHBDF$

       000000                       ANBDF$                      ; DEFINE ALIAS NAME BLOCK OFFSETS
       000000                       MANDF$                      ; DEFINE NETWORK MANAGEMENT SYMBOLS
       000000                       HDRDF$                      ; DEFINE TASK HEADER OFFSETS
       000000                       DHBDF$                      ; DEFINE DEC HOME BLOCK OFFSETS

       000000                       ALRCX$  LIST                ; DEFINE CONTEXT AREA OFFSETS
                               ;
                               ; CONTEXT AREA OFFSET DEFINITIONS
                               ;
               000020               MAXID=16.                   ; MAXIMUM LENGTH OF USER ID
               000010               MAXPAS=8.                   ; MAXIMUM LENGTH OF PASSWORD
               000020               MAXACC=16.                  ; MAXIMUM LENGTH OF ACCOUNT
       000020                       .ASECT
               000000               .=0
                               ;
                               ; OFFSETS IN COMMON WITH CHANGE INFORMATION MODULES
                               ;
       000000                  A$FLG:  .BLKW   1                ; FLAG WORD
       000002                  A$OPT:  .BLKW   1                ; SAVED OPTIONS WORD
       000004                  A$PAR:  .BLKW   1                ; CURRENT PARAMETER TYPE
       000006                  A$BUF:  .BLKW   1                ; SAVED POINTER TO BUFFER
       000010                  A$NUM:  .BLKB   1                ; NUMBER OF BYTES IN ANB (FOR SET)
       000011                  A$TYP:  .BLKB   1                ; ALIAS FORMAT TYPE
       000012                  A$ALI:  .BLKW   3                ; ALIAS NAME, SPACE FILLED
       000020                  A$SCO:  .BLKW   1                ; SCOPE OF ALIAS
                               ;
                               ; OFFSETS UNIQUE TO READ INFORMATION PARAMETERS MODULES
                               ;
       000022                  A$NEX:  .BLKW   1                ; POINTER TO NEXT ALIAS TO BE DISPLAYED
       000024                  A$LEN:                           ; LENGTH OF CONTEXT AREA
       000024                  A$SAV:                           ; SAVE AREA FOR NAMES
       000000                          .PSECT
                               ;
                               ; FLAGS WORD BIT DEFINITIONS
                               ;
               000001                  AF.INI=1                 ; SUCCESSFUL INITIALIZATION COMPLETED
               000002                  AF.OVR=2                 ; CONTEXT AREA OVERFLOWED
```

```
                                    .SBTTL  ALREIN - ALIAS READ INFORMATION INITIALIZE
```

```
**FILE**ID**ALREOP







  AAAAAA     LL           RRRRRRRR     EEEEEEEEEE     OOOOOO     PPPPPPPP
  AAAAAA     LL           RRRRRRRR     EEEEEEEEEE     OOOOOO     PPPPPPPP
AA      AA   LL           RR      RR   EE             OO      OO   PP      PP
AA      AA   LL           RR      RR   EE             OO      OO   PP      PP
AA      AA   LL           RR      RR   EE             OO      OO   PP      PP
AA      AA   LL           RR      RR   EE             OO      OO   PP      PP
AA      AA   LL           RRRRRRRR     EEEEEEEE       OO      OO   PPPPPPPP
AA      AA   LL           RRRRRRRR     EEEEEEEE       OO      OO   PPPPPPPP
AAAAAAAAAA   LL           RR  RR       EE             OO      OO   PP
AAAAAAAAAA   LL           RR  RR       EE             OO      OO   PP
AA      AA   LL           RR    RR     EE             OO      OO   PP                ....
AA      AA   LL           RR    RR     EE             OO      OO   PP                ....
AA      AA   LLLLLLLLLL   RR      RR   EEEEEEEEEE     OOOOOO     PP                ....
AA      AA   LLLLLLLLLL   RR      RR   EEEEEEEEEE     OOOOOO     PP                ....






LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06
Table of contents
```

MACRO CALLS AND LOCAL DATA

```
                                                    H  3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-2
Symbol table

MD.ROB  000005      MF$ALL  177775      MO$SET  000000      MP$CUS  002127      MP$LMB  002152
MD.RSH  000013      MF$BYE  000300      MO$STA  000020      MP$CVA  000540      MP$LMX  002200
MD.SDU  000005      MF$CHA  000023      MO$SUM  000000      MP$DAL  002570      MP$LNA  000144
MD.SLD  000003      MF$DUM  000020      MO$TSK  000000      MP$DCO= 001464      MP$LNO  000233
MD.SNA  000000      MF$EVT  000001      MO$VOL  000000      MP$DDT  002177      MP$LOA  000170
MD.TLO  000004      MF$FUP  000264      MO$ZER  000000      MP$DEL  001131      MP$LOG  004514
MD.UNA  000002      MF$KNO  177777      MP$ACB  000012      MP$DES  000156      MP$LOO  000620
MD.UNR  000003      MF$LOA  000017      MP$ACC  000514      MP$DEV  002114      MP$LTM  001613
MD.UOB  000007      MF$LOO  177775      MP$ACT  004526      MP$DFA  001320      MP$LTY  004533
MD.VOL  000006      MF$REA  000024      MP$ADD  000766      MP$DHO= 001465      MP$LWI  000230
ME$ALI  000010      MF$SIG  177773      MP$ADJ  001440      MP$DIA  000173      MP$MAC  004420
ME$AL2  000003      MF$SPF  000302      MP$ADP  006325      MP$DLB  002571      MP$MAD  001630
ME$ARE  000005      MF$SYS  000026      MP$ADS  004406      MP$DLG  004521      MP$MAP  006323
ME$CIR  000003      MF$TES  000022      MP$ALB  000036      MP$DLI= 001466      MP$MAR  001635
ME$EXA  000000      MF$TRA  000301      MP$AMC  001640      MP$DLT  002200      MP$MAV  006321
ME$EXE  000200      MF$TRI  000021      MP$AMH  001641      MP$DRO  001441      MP$MBN  001636
ME$LIN  000001      MF$ZER  000025      MP$ANB  004420      MP$DST  000454      MP$MBR  001637
ME$LOG  000002      ML$ALL  100000      MP$ASB  000024      MP$DTE  002140      MP$MBU  001642
ME$MOD  000004      ML$CLS  000000      MP$ASC  004432      MP$DTY= 001452      MP$MCB  000156
ME$NOD  000000      ML$CON  000001      MP$AUS  000512      MP$DUA  000207      MP$MCO  001632
ME$NON  177777      ML$EXT  000001      MP$BBT  002165      MP$DUC  000210      MP$MDE  004421
ME$OBJ  000007      ML$FIL  000002      MP$BDF  002164      MP$DUM  000202      MP$MHO  001633
ME$OB2  000004      ML$FIR  000000      MP$BFQ  002121      MP$DUP  002127      MP$MLB  000202
ME$PRO  000005      ML$INT  000000      MP$BLK  001616      MP$DVC  002114      MP$MLK  001306
ME$SYS  000006      ML$KNO  140000      MP$BLO  001452      MP$DWE  001321      MP$MLN  001631
ME.BLO  177744      ML$MON  000003      MP$BMX  002176      MP$ELT  000157      MP$MLP  006333
ME.CDN  177753      ML$SYS  000002      MP$BNP  006330      MP$ETY  001605      MP$MRB  002171
ME.CST  177765      ML$TOP  000001      MP$BRT  001620      MP$EVE  000311      MP$MRO  001605
ME.DIS  177755      MN$UNL  000377      MP$BSA  002176      MP$FNC  001752      MP$MRP  006322
ME.DON  177600      MO$ACC  000200      MP$BSD  002203      MP$GDT  002222      MP$MRT  002153
ME.FCO  177762      MO$ADD  000002      MP$BSI  002200      MP$GNM  002223      MP$MRV  006320
ME.FIO  177756      MO$ALA  000004      MP$BSP  006331      MP$GRO  002115      MP$MSB  000170
ME.FOP  177763      MO$ALI  000000      MP$BUF  000170      MP$GRP  000541      MP$MVE  000145
ME.FOR  177776      MO$CHA  000040      MP$BUP  006324      MP$GTY  002224      MP$MVI  001634
ME.FUN  177777      MO$CIR  000003      MP$BUS  001643      MP$HAD  001757      MP$MVR  001751
ME.GRO  177745      MO$CLE  000100      MP$CAC  001750      MP$HBT  002142      MP$MWN  002154
ME.HAR  177750      MO$COU  000060      MP$CAS  001762      MP$HDD  000162      MP$MXB  002172
ME.IID  177767      MO$DAC  000000      MP$CAT  002210      MP$HTM  001612      MP$MXC  000466
ME.LCO  177766      MO$DEF  000000      MP$CCS  004406      MP$HWA  002210      MP$MXR  001630
ME.LPR  177757      MO$DPR  000000      MP$CHN  002141      MP$IAT  001322      MP$MXW  002143
ME.MPR  177773      MO$ENT  000017      MP$CIR  000144      MP$IDE  000144      MP$NAA  006335
ME.MVE  177771      MO$EVE  000100      MP$CLK  002131      MP$IDP  006327      MP$NAC  001130
ME.OPE  177747      MO$INF  000160      MP$CLN  002126      MP$IHO  000215      MP$NAP  006326
ME.PLO  177751      MO$INS  000002      MP$CLT  002211      MP$INA  002177      MP$NCT  000240
ME.PMI  177743      MO$LIN  000001      MP$CMB  002142      MP$IND  002204      MP$NET  002114
ME.PNA  177752      MO$LOG  000001      MP$CMK  000537      MP$INI  002201      MP$NLI  000765
ME.PRI  177775      MO$MIR  000031      MP$CMX  002153      MP$ITI  000776      MP$NNA  000764
ME.PTY  177772      MO$NAM  000001      MP$CND  000310      MP$LAA  000231      MP$NOD  000500
ME.PVA  177760      MO$NIC  000023      MP$CNU  001753      MP$LAN  000234      MP$NRA  006334
ME.RES  177761      MO$NOD  000000      MP$COB  000311      MP$LAR  000202      MP$NRB  000214
ME.ROO  177754      MO$OFF  000001      MP$CON  002126      MP$LAS  000012      MP$NSA  001617
ME.SIZ  177774      MO$ON   000000      MP$COS  001604      MP$LBS  000203      MP$NTI  002141
ME.SYS  177746      MO$OPT  000001      MP$CPF  000230      MP$LCO  000226      MP$NUM  001642
ME.UCC  177770      MO$PER  000200      MP$CPL  000232      MP$LCT  000156      MP$NVE  001274
MF$ACT  177776      MO$PRO  000002      MP$CPT  000226      MP$LHL  000232      MP$NXN= 001476
MF$ADD  000000      MO$PRS  000001      MP$CPU  000161      MP$LLE  000227      MP$OAC  000632
MF$ADJ  177774      MO$REA  000200      MP$CSZ  001755      MP$LLO  004432      MP$OAN  000620
```

```
                                                    I  3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-3
Symbol table

MP$OCO  000776      MP$SGZ  001644      MS$ATR  000011      MT$NLF  000017      M$$FCS= 000000
```

```

                                        .ENDC   ; NDF R$$PRO

000304  120527  000001                  CMPB    R5,#MS$TER              ; ELSE, IS IT A LEGAL SCOPE TYPE ?
000310  001102                          BNE     30$                     ; IF NE, NO - ERROR
                                  ;
                                  ; IF SCOPE TYPE = MS$TER (TERMINAL ID STRING), GET UCB ADDRESS
                                  ;
000312  005302                          DEC     R2                      ; DEC BUFFER COUNT FOR SCOPE COUNT
000314  112305                          MOVB    (R3)+,R5                ; GET LENGTH OF SCOPE STRING
000316  160502                          SUB     R5,R2                   ; DEC BUFFER COUNT BY SCOPE STRING LENGTH
000320  002476                          BLT     30$                     ; IF LT, ERROR
000322  020527  000003                  CMP     R5,#3                   ; CHECK MINIMUM LENGTH OF SCOPE STRING
000326  002473                          BLT     30$                     ; IF LT, TOO SMALL - ERROR

                                        .IF DF  R$$PRO  ; PRO/DECNET

                                        CMPB    (R3),#'T                ; IS SCOPE TI: ?
                                        BNE     30$                     ; IF NE, NO - ERROR

                                        .ENDC   ; DF R$$PRO

000330  112364  000020                  MOVB    (R3)+,A$SCO(R4)         ; SAVE FIRST DEVICE NAME CHAR

                                        .IF DF  R$$PRO  ; PRO/DECNET

                                        CMPB    (R3),#'I                ; IS SCOPE TI: ?
                                        BNE     30$                     ; IF NE, NO - ERROR

                                        .ENDC   ; DF R$$PRO

000334  112364  000021                  MOVB    (R3)+,A$SCO+1(R4)       ; SAVE SECOND DEVICE NAME CHAR
000340  010300                          MOV     R3,R0                   ; POINT TO ASCII UNIT NUMBER FOR $COTB
000342  010246                          MOV     R2,-(SP)                ; SAVE R2
000344                                  CALL    $COTB                   ; CONVERT OCTAL STRING TO BINARY
000350  012602                          MOV     (SP)+,R2                ; RESTORE R2
000352  010003                          MOV     R0,R3                   ; POINT TO NEXT CHARACTER IN BUFFER
000354  126327  177777  000072          CMPB    -1(R3),#':              ; IS THERE A LEGAL DEVICE TERMINATOR ?
000362  001055                          BNE     30$                     ; IF NE, NO - ERROR
000364                                  CALL    DEALUN                  ; DEASSIGN THE TEMPORARY LUN FIRST
                                                                        ; (AVOIDS RSX ''LUN LOCKED IN USE ERROR)
000370                                  ALUN$S  #$TMLUN,A$SCO(R4),R1    ; TRY TO ASSIGN A LUN TO THE DEVICE
000410  103442                          BCS     30$                     ; IF CS, WE COULDN'T - ERROR

000412                                  SWSTK$  20$                     ;; ELSE, ENTER SYSTEM STATE TO GET UCB

                                        .IF     DF,R$$MPL

                                        SAVMAP                          ;; SAVE CURRENT KISAR6 MAPPING
                                        MOV     @SAHDB,@KSAR6           ;; MAP TO MY EXTERNAL HEADER
                                        MOV     @SAHPT,R0               ;; POINT TO THE HEADER
                                        MOV     <H.LUN+<4*<$TMLUN-1>>>(R0),R1 ;; GET THE DEVICE'S UCB ADDRESS
                                        RESMAP                          ;; RESTORE PREVIOUS KISAR6

                                        .IFF    ; DF R$$MPL

000416  017700  000000G                 MOV     @HEADR,R0               ;; GET OUR HEADER'S ADDRESS     ; RJK02
```

```
000422  016001  000000G                 MOV     <H.LUN+<4*<$TMLUN-1>>>(R0),R1 ;; GET THE DEVICE'S UCB ADDRESS
```

```
     49                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     50                                  ;
     51                                  ; LIBRARY MACRO CALLS
     52                                  ;
     53
     54                                          .MCALL  MANDF$,XPDDB$,PDVDF$,RETC,ARRCX$,UCBDF$
     55                                          .MCALL  NSFDF$,HDRDF$,DHBDF$,RESMAP,SAVMAP
     56
     57 000000                                   DHBDF$                          ; DEFINE DECNET HOME BLOCK
     58 000000                                   HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     59 000000                                   MANDF$                          ; DEFINE NETWORK MANAGEMENT FLAGS
     60 000000                                   NSFDF$                          ; DEFINE NSP FEATURES MASK
     61 000000                                   PDVDF$                          ; DEFINE PDV OFFSETS
     62 000000                                   UCBDF$                          ; DEFINE UCB OFFSETS
     63 000000                                   XPDDB$                          ; DEFINE XPT DATABASE OFFSETS
     64
     65 000000                                   ARRCX$  LIST                    ; DEFINE CONTEXT AREA
                                         ;
                                         ; CONTEXT AREA DEFINITION
                                         ;
        000020                                   .ASECT
                       000000                    . = 0
        000000                           A$FLG:  .BLKW   1                       ; FLAGS WORD
        000002                           A$BUF:  .BLKW   1                       ; OUTPUT BUFFER ADDRESS
        000004                           A$NUM:  .BLKW   1                       ; AREA NUMBER
        000006                           A$MAX:  .BLKW   1                       ; HIGHEST AREA IN DATABASE
        000010                           A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
        000012                           A$XPT:  .BLKW   1                       ; XPT PDV ADDRESS
        000014                           A$EXA:  .BLKW   1                       ; EXECUTOR AREA
        000016                           A$PAR:  .BLKW   1                       ; PARAMETER FOR RESPONSE MESSAGE
        000020                           A$NAM:  .BLKB   7                       ; NODENAME BUFFER
        000027                           A$LENX:
        000000                                   .PSECT
                                         ;
                                         ; FLAGS WORD BIT DEFINITIONS
                                         ;
                       000001                    AF.INI  =       1               ; INITIALIZATION FLAG
                       000002                    AF.KNO  =       2               ; KNOWN/ACTIVE/SIGNIFIGANT AREAS
                       000020                    AF.MUX  =       20              ; CIRCUIT IS MUX
                       000040                    AF.END  =       40              ; EXECUTOR IS AN ENDNODE
                       000100                    AF.LV2  =       100             ; EXECUTOR IS AN AREA ROUTER
                       000077                    MXAREA  =       63.             ; MAXIMUM AREA NUMBER
     66
```

```
     68                                          .SBTTL  ARREIN - READ AREA STATUS INITIALIZATION
```

Table of contents

```
     5-   51     MACRO CALLS AND LOCAL DATA
     6-   75     ARREOP - AREA READ STATUS OPERATE
     7-  189     GETEXA - GET EXECUTOR AREA INFORMATION
     8-  252     GETMHC - GET AREA MINIMUM HOPS AND COST
     9-  315     FMTCIR - FORMAT CIRCUIT ID
    10-  400     NXTNOD - STORE THE NEXT NODE
    11-  426     NODNAM - FIND AND FORMAT NODE NAME
```

```
      1                                             .TITLE  ARREOP - AREA READ INFORMATION OPERATE
```

```
                                                H  7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-1
Symbol table

AF.END= 000040            D$IPL   000051            LF.PAC= 000200            MC$CDC  002045            MC$TBR  001140
AF.INI= 000001            D$LID   000020            LF.RDY= 040000            MC$CDP  001441            MC$TBS  001141
AF.KNO= 000002            D$LNAM  000006            LF.REA= 010000            MC$CIF  001465            MC$TDR  002053
AF.LV2= 000100            D$LNUM  000014            LF.SER= 000040            MC$CLD  001464            MC$TIM  000000
AF.MUX= 000020            D$LST   000047            LF.TIM= 000010            MC$COU  100000            MC$TYP  007777
ARREOP  000002RG          D$MAXC  000064            LF.UNL= 020000            MC$CTL  001454            MC$UBU  002052
ASCIMG= 000100            D$MAXH  000066            LF.X2P= 000000            MC$CTR  001452            MC$UFD  002047
AT$ACL= 000100            D$MAXV  000070            LN.CLO= 000000            MC$CTS  001453            MC$UMR  001132
AT$ADN= 000040            D$MLL   000040            LN.DUM= 000005            MC$DOV  002050            MC$UMS  001133
AT$AUP= 000020            D$MNOD  000041            LN.LOA= 000004            MC$LBR  001750            MC$VAX  000003
AT$CYC= 000004            D$NA    000062            LN.LOO= 000003            MC$LBS  001751            MC$WID  060000
AT$LV1= 000002            D$NBEA  000056            LN.OAU= 000003            MC$LDI  001774            MC$WIH  040000
AT$LV2= 000001            D$NBRA  000054            LN.OFF= 000001            MC$LDO  001775            MC$WIL  020000
AT$NEX= 000200            D$NEND= 000054            LN.ON  = 000000           MC$LDR  001762            MC$W08  020000
AT$UP  = 000010           D$NLN   000030            LN.ODP= 000004            MC$LDS  001763            MC$W16  040000
A$BUF   000002            D$NN    000060            LN.OPE= 000001            MC$LLB  002021            MC$W32  060000
A$CIR   000003            D$OUTT  000043            LN.REF= 000002            MC$LLP  002115            MC$XBR  001750
A$EXA   000014            D$RETF  000050            LN.SER= 000002            MC$LLR  002007            MC$XBS  001751
A$FLG   000000            D$RNN   000002            LN.STA= 000017            MC$LOO  000001            MC$XCJ  000322
A$LEN   000004            D$RTMR  000076            LN.SUB= 000360            MC$LRB  002020            MC$XCR  002260
A$LENX  000027            D$SEG   000036            LN.TRI= 000006            MC$LRP  002114            MC$XCS  002261
A$MAX   000006            D$SER   000032            L$$ASG= 000000            MC$LRR  002006            MC$XDR  001762
A$NAM   000020            D$SQRL  000052            L$$DRV= 000000            MC$LSI  002032            MC$XDS  001763
A$NID   000000            D$$BUG= 177514            L$$P11= 000001            MC$LST  002033            MC$XFR  002272
A$NUM   000004            D$$ISK= 000000            L$$11R= 000000            MC$LUP  006072            MC$XFS  002273
A$OPT   000010            D$$L11= 000001            L.COST  000015            MC$MAP  010000            MC$XLJ  000323
A$PAR   000016            D$$YNC= 000000            L.CTL   000012            MC$MBL  001764            MC$XLR  002330
A$TM    000003            D$$YNM= 000000            L.CVA   177776            MC$MBX  004420            MC$XMA  000310
A$TMI   000002            E$$XPR= 000000            L.DDM   000002            MC$MBY  001752            MC$XMC  002305
A$TSZ   000000            FMTCIR  001054R           L.DDS   000004            MC$MPX  004406            MC$XMS  002304
A$TYP   000002            F$$LVL= 000001            L.DLC   000003            MC$NAP  001604            MC$XNR  002332
A$XPT   000012            GETEXA  000416R           L.DLM   000006            MC$NBR  001130            MC$XRC  002316
A$$CHK= 000000            GETMHC  000650R           L.DLS   000010            MC$NBS  001131            MC$XRR  002331
A$$CPS= 000000            G$$TPP= 000000            L.FLG   000000            MC$NCR  001154            MC$XRS  002342
A$$PRI= 000000            G$$TSS= 000000            L.KRBA  000016            MC$NCS  001155            MC$020  000002
A$$TRP= 000000            G$$TTK= 000000            L.LEN  = 000022           MC$NML  001274            MD$CI   000007
CODMUL= 000300            G$$WRD= 000000            L.MPF   000022            MC$NMR  001142            MD$CNA  000003
CODSNG= 000200            HEX    = 000040           L.NMST  000020            MC$NMS  001143            MD$COU  000001
C$$CKP= 000000            HEXIMG= 177777            L.NSTA  000014            MC$NNO  001606            MD$DA   000010
C$$ORE= 000400            I$$RAR= 000000            L.DWNR  000021            MC$NNU  001605            MD$DL   000004
C$$RSH= 177564            I$$RDN= 000000            L.UNT   000013            MC$NDP  001607            MD$DLV  000020
DECPT  = ****** GX        KISAR6= ****** GX         MAPADD= ****** GX         MC$NOR  000000            MD$DMC  000014
DECSIG= 000020            K$$CNT= 177546            MB$DIS  000001            MC$NPF  001616            MD$DMF  000046
DECUNS= 000000            K$$CSR= 177546            MB$ENA  000000            MC$NPR  001630            MD$DMP  000022
D$AMXC  000072            K$$LDC= 000000            MB$FUL  000002            MC$NRE  001200            MD$DMR  000050
D$AMXH  000074            K$$TPS= 000074            MB$MIX  000002            MC$NRT  001166            MD$DMV  000042
D$ANN   000000            LC$EXT  000000            MB$ONE  000001            MC$NVR  001642            MD$DN   000016
D$BRPR  000102            LC$INT  000001            MB$RXO  000001            MC$P08  000000            MD$DP   000000
D$BRTM  000100            LD$LP  = 000000           MB$TXO  000000            MC$P11  000001            MD$DPV  000044
D$DELF  000045            LF.ACT= 100000            MB$ZER  000000            MC$RFL  002046            MD$DQ   000006
D$DELW  000046            LF.BRO= 000400            MC$BAB  006073            MC$SBU  002051            MD$DTE  000024
D$END  = 000104           LF.BWT= 000007            MC$BID  001765            MC$SFL  002044            MD$DU   000002
D$FNB   000034            LF.ENA= 002000            MC$BMC  001767            MC$SYC  000012            MD$DUP  000012
D$HIOR  000024            LF.LPB= 001000            MC$BSC  001766            MC$SYL  000036            MD$DV   000026
D$HOST  000022            LF.MDC= 000100            MC$CAC  001442            MC$SYR  000050            MD$DZ   000030
D$INAC  000044            LF.MFL= 004000            MC$CAP  001440            MC$SYS  000024            MD$FUL  000000
D$INCT  000042            LF.MTP= 000020            MC$CCL  001445            MC$SYZ  000000            MD$HAL  000001
```

```
                                                I  7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-2
Symbol table
```

```
                                                                                        H  8
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 5-1
MACRO CALLS AND LOCAL DATA

        000032                        N$HREM: .BLKW   1                      ; HIGHEST REMOTE NODE ADDRESS
        000034                        N$HROL: .BLKW   1                      ; HIGHEST ROL NODE ADDRESS
        000036                        N$HECL: .BLKW   1                      ; HIGHEST ECL NODE ADDRESS
        000040                        N$HADJ: .BLKW   1                      ; HIGHEST ADJACENCY NODE ADDRESS
        000042                        N$HSER: .BLKW   1                      ; HIGHEST SERVICE NODE ADDRESS
        000044                        N$HLOO: .BLKW   1                      ; HIGHEST LOOP NODE ADDRESS
        000046                        N$HLCX: .BLKW   1                      ; HIGHEST NODE ADDRESS LAST CONTEXT AREA
        000050                        N$HCCX: .BLKW   1                      ; HIGHEST NODE ADDRESS CURRENT CONTEXT AREA
        000052                        N$CLOO: .BLKW   1                      ; HIGHEST LOOP NODE ADDRESS CURRENT CONTEXT AREA
        000054                        N$LLOO: .BLKW   1                      ; HIGHEST LOOP NODE ADDRESS LAST CONTEXT AREA
        000056                        N$BUF:  .BLKW   1                      ; OUTPUT BUFFER ADDRESS
        000060                        N$BFL:  .BLKW   1                      ; BUFFER LIMIT OF SAVED NODE ADDRESSES
        000062                        N$NEX:  .BLKW   1                      ; POINTER TO NEXT NODE TO BE DISPLAYED
        000064                        N$SAV:                                 ; SAVE AREA FOR NODE ADDRESSES
                000056                        .=N$BUF
        000056                        N$AMX:  .BLKW   1                      ; MAXIMUM AREA NUMBER IN BITMASK
        000060                        N$NMX:  .BLKW   1                      ; MAXIMUM NODE ADDRESS IN BITMASK
        000062                        N$CAR:  .BLKW   1                      ; CURRENT AREA NUMBER IN SCAN
        000064                        N$CNO:  .BLKW   1                      ; CURRENT NODE ADDRESS IN SCAN
        000066                        N$AMP:  .BLKW   1                      ; BITMAP OF AREAS WITH NODES
                000002                N$ALN=.-N$AMP
        000070                        N$NMP:  .BLKW   64.                    ; BITMAP OF NODES IN CURRENT AREA
                000200                N$NLN=.-N$NMP
        000270                        N$LOO:                                 ; LOOP NODE CHANNEL NUMBERS
        000270                        N$LEN:                                 ; LENGTH NEEDED FOR CONTEXT AREA

        000000                                .PSECT
                                      ;
                                      ; FLAGS WORD BIT DEFINITIONS
                                      ;
                000001                        NF.INI = 1                     ; SUCCESSFUL INITIALIZATION
                000002                        NF.OVR = 2                     ; CONTEXT AREA OVERFLOWED
                000004                        NF.LPB = 4                     ; SET NODE IS A LOOPBACK (NAME SPECIFIED)
                000010                        NF.MUX = 10                    ; LINE ID DEVICE IS A MULTIPLEXER
                000020                        NF.REA = 20                    ; NODE IS REACHABLE
                000040                        NF.LV1 = 40                    ; LEVEL 1 ROUTING TASK PRESENT
                000100                        NF.LV2 = 100                   ; LEVEL 2 ROUTING TASK PRESENT
                000200                        NF.ARE = 200                   ; NODE IS OUTSIDE EXECUTOR AREA
                000400                        NF.VR2 = 400                   ; CONNECTED TO NICE V2.0 NCP
                001000                        NF.VR3 = 1000                  ; CONNECTED TO NICE V3.0 NCP
                002000                        NF.EPM = 2000                  ; EPM IS PRESENT
                004000                        NF.LXX = 4000                  ; LOOP NODE FLAG BIT
                010000                        NF.PA2 = 10000                 ; SECOND PASS OF REMOTE NODE SCAN
                020000                        NF.EXN = 20000                 ; EXECUTOR NODE IS IN BITMAP
                040000                        NF.FTT = 40000                 ; FIRST TIME THROUGH NODE LIST
                                      ;
                                      ; LIST FLAG WORD BIT DEFINITIONS (N$LFLG)
                                      ;
                000001                        NL.REM = 1                     ; REMOTE NODE LIST
                000002                        NL.ROL = 2                     ; ROL NODE LIST
                000004                        NL.ECL = 4                     ; ECL NODE LIST
                000010                        NL.ADJ = 10                    ; ADJACENCY NODE LIST
                000020                        NL.SER = 20                    ; SERVICE NODE LIST
                000040                        NL.LOO = 40                    ; LOOP NODE LIST
     86
     87 000000                                CIRCX$  LIST
                                      ;
```

```
                                                                                        I  8
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 5-2
MACRO CALLS AND LOCAL DATA

                                      ; CONTEXT AREA OFFSET DEFINITIONS
```

```
                                        .SBTTL  SAVCTX - SAVE NODE CONTEXT
                                ;+
                                ; *** - SAVCTX - SAVE NODE CONTEXT
                                ;
                                ; INPUTS:
                                ;       R1 = ADDRESS OF NODE TO SAVE
                                ;       R4 = ADDRESS OF CONTEXT AREA
                                ;       R5 = NEXT AVAILABLE LOCATION IN CONTEXT SAVE AREA
                                ;
                                ; OUTPUTS:
                                ;       R5 = NEXT FREE WORD IN CONTEXT SAVE AREA
                                ;       C-BIT =
                                ;          SET:   BIT NF.OVR SET IN N$FLAG(R4) IF CONTEXT AREA FULL
                                ;          CLEAR: BIT NF.OVR CLEAR IN N$FLAG(R4) IF CONTEXT AREA NOT FULL
                                ;
                                ;       NODES ARE SAVED AS 2 WORD ENTRIES SORTED BY ADDRESS:
                                ;
                                ;        WORD 0  - NUMBER OF NODES IN SEQUENCE (0 IS RESERVED)
                                ;        WORD 1  - 16 BIT NODE ADDRESS OF FIRST NODE IN SEQUENCE
                                ;
                                ;-

001262                          SAVCTX:
001262                                  CALL    $SAVAL                  ; SAVE EVERYTHING
001266  032764  000004  000000          BIT     #NF.WLD,N$FLAG(R4)      ; WILDCARD NODE ADDRESS SPECIFIED ?
001274  001406                          BEQ     2$                      ; IF EQ, NO
001276  010103                          MOV     R1,R3                   ; ELSE COPY NODE ADDRESS
001300  042703  001777                  BIC     #1777,R3                ; ISOLATE AREA
001304  020364  000004                  CMP     R3,N$ADD(R4)            ; SAME AREA AS WILDCARD NODE ?
001310  001141                          BNE     90$                     ; IF NE, DON'T SAVE IT

001312  016403  177776          2$:     MOV     -2(R4),R3               ; ELSE GET SIZE OF CONTEXT AREA
001316  060403                          ADD     R4,R3                   ; POINT PAST END OF CONTEXT AREA
001320  024343                          CMP     -(R3),-(R3)             ; POINT TO LAST AVAILABLE ENTRY
001322  020503                          CMP     R5,R3                   ; ROOM FOR ANOTHER ENTRY ?
001324  103403                          BLO     3$                      ; IF LO, YES
001326  052764  000002  000000          BIS     #NF.OVR,N$FLAG(R4)      ; ELSE SET OVERFLOW BIT

001334  032701  176000          3$:     BIT     #^C<1777>,R1            ; AREA SPECIFIED ?
001340  001005                          BNE     5$                      ; IF NE, YES
001342  016446  000026                  MOV     N$NSP(R4),-(SP)         ; ELSE PICK UP EXEC ADDRESS
001346  042716  001777                  BIC     #<1777>,(SP)            ; ISOLATE AREA BITS
001352  052601                          BIS     (SP)+,R1                ; USE OUR AREA
001354  020164  000026          5$:     CMP     R1,N$NSP(R4)            ; IS THIS THE EXECUTOR NODE ?
001360  001515                          BEQ     90$                     ; IF EQ, ALREADY SAVED - DONE
001362  010403                          MOV     R4,R3                   ; COPY CONTEXT AREA ADDRESS
001364  062703  000070                  ADD     #N$SAV+4,R3             ; POINT SAVED NODE (AFTER EXECUTOR)

001370  020305                  10$:    CMP     R3,R5                   ; END OF SAVED NODE DESCRIPTIONS ?
001372  103073                          BHIS    70$                     ; IF HIS, YES - ADD THIS NODE
001374  012300                          MOV     (R3)+,R0                ; PICK UP SET SIZE
001376  012302                          MOV     (R3)+,R2                ; PICK UP STARTING NODE ADDRESS
001400  020102                          CMP     R1,R2                   ; SEE IF LESS THAN THIS SET
001402  001504                          BEQ     90$                     ; IF EQ, ALREADY SAVED
001404  101010                          BHI     20$                     ; IF HI, CHECK HIGH BOUND
                                ;
                                ; CHECK LOW BOUND OF CURRENT ENTRY
```

```
BLNLST     CREATED BY  MACRO  ON 29-JUN-85 AT 12:09     PAGE 6
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE               REFERENCES

ZF.DIA   = 004000            #5-69
ZF.DLC   = 000002            #5-69
ZF.DVP   = 100000            #5-69
ZF.INI   = 040000            #5-69
ZF.KMX   = 000020            #5-69
ZF.LLC   = 000004            #5-69
ZF.LMC   = 000100            #5-69
ZF.MAN   = 020000            #5-69
ZF.MFL   = 000010            #5-69
ZF.MTM   = 000400            #5-69
ZF.MUX   = 000040            #5-69
ZF.PSE   = 002000            #5-69
ZF.SLI   = 010000            #5-69
ZF.TIM   = 000200            #5-69
ZF.X3P   = 000000            #5-69
ZS.ASN   = 100000            #5-69
ZS.BSY   = 140000            #5-69
Z.AVL      000014            #5-69
Z.DAT      000016            #5-69
Z.DSP      000000            #5-69        5-69
Z.FLG      000010            #5-69
Z.LEN    = 000016            #5-69
Z.LLN      000006            #5-69
Z.MAP      000020            #5-69
Z.NAM      000004            #5-69
Z.PCB      000012            #5-69
Z.SCH      000007            #5-69
$CEACC   = ******   GX        15-447
$SAVAL   = ******   GX        16-482
```

```
BLNLST     CREATED BY  MACRO  ON 29-JUN-85 AT 12:09     PAGE 7
MACRO CROSS REFERENCE                                    CREF   04.00
```

```
                                                          H 11
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE             REFERENCES

BIT0    = 000001          #6-63
BIT1    = 000002          #6-63
BIT10   = 002000          #6-63
BIT11   = 004000          #6-63
BIT12   = 010000          #6-63
BIT13   = 020000          #6-63
BIT14   = 040000          #6-63
BIT15   = 100000          #6-63
BIT2    = 000004          #6-63
BIT3    = 000010          #6-63
BIT4    = 000020          #6-63
BIT5    = 000040          #6-63
BIT6    = 000100          #6-63
BIT7    = 000200          #6-63
BIT8    = 000400          #6-63
BIT9    = 001000          #6-63
B$10      000003          #5-63
B$11      000004          #5-63
B$12      000005          #5-63
B$5       000000          #5-63
B$6       000001          #5-63
B$7       000002          #5-63
B$7S10    000006          #5-63
CHKCIR    000000 RG       #7-87
LF$END  = 000020          #6-63
LF$MLT  = 040000          #6-63
LF$REA  = 000001          #6-63
LF$SEG  = 100000          #6-63
LF$SIG  = 000040          #6-63
LF$SKP  = 000004          #6-63
LF$VR2  = 000010          #6-63
LF$ZER  = 000002          #6-63
LP$MPT  = 000010          #6-63
LP$MUX  = 000004          #6-63
LP$TRB  = 000002          #6-63      7-95
LP$UNT  = 000001          #6-63      7-95
LP$WCN  = 000040          #6-63      6-63       7-95
LP$WDV  = 000020          #6-63      6-63       7-95
LP$WLD  = 000360          #6-63
LP$WTR  = 000200          #6-63      6-63       7-95
LP$WUN  = 000100          #6-63      6-63       7-95
L$ADJ     000110          #5-63
L$BUF     000060          #5-63
L$CHN     000054          #5-63
L$COU     000114          #5-63
L$CTB     000053          #5-63
L$CTL     000002          #5-63
L$DDM     000000          #5-63
L$FLG     000064          #5-63
L$FLX     000070          #5-63
L$LEN     000124          #5-63
L$LTM     000070          #5-63
```

```
                                                          I 11
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00
```

```
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42  Page 7-4
Symbol table

SY:CHKLIN.V2,[135,134]CHKLIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CHKLIN
```

```
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
                                                                     H 13
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 8
SET CIRCUIT PARSE TABLE

     80                                        .SBTTL  SET CIRCUIT PARSE TABLE
     81
     82 000000                                 IMSG$   SETCIR
     83                                 ;
     84                                 ;      STATES FOR SET CIRCUIT OPTIONS
     85                                 ;
     86 000000                                 FIELD$
     87                                 .IF NDF R$$11S
     88 000000                                  $SBEXP SALL,END
     89                                 .IFTF
     90 000000                                  $SBEXP SSTA,,,LS$STA,+L$FLG
     91                                 .IFT
     92 000000                                  $NEXT
     93 000016                                 FIELD$
     94 000000                                  $SBEXP SSER,,,LS$SER,+L$FLG            ; Service
     95 000000                                  $NEXT
     96 000026                                 FIELD$
     97 000000                                  $SBEXP SLCT,,,LS$LCT,+L$FLX            ; Counter timer
     98 000000                                  $NEXT
     99 000036                                 FIELD$
    100 000000                                  $SBEXP SCOS,END,,LS$COS,+L$FLG         ; Cost
    101 000000                                  $NEXT
    102 000050                                 FIELD$
    103 000000                                  $SBEXP SHTM,,,LS$HTM,+L$FL1            ; Hello timer
    104 000000                                  $NEXT
    105 000060                                 FIELD$
    106 000000                                  $SBEXP SLTM                            ; (Listen timer)
    107 000000                                  $NEXT
    108 000064                                 FIELD$
    109 000000                                  $SBEXP SBLK,,,LS$BLK,+L$FLX            ; Blocking
    110 000000                                  $NEXT
    111 000074                                 FIELD$
    112 000000                                  $SBEXP SMXR,,,LS$MXR,+L$FLX            ; Maximum recalls
    113 000000                                  $NEXT
    114 000104                                 FIELD$
    115 000000                                  $SBEXP SRET,,,LS$RET,+L$FLX            ; Recall timer
    116 000000                                  $NEXT
    117 000114                                 FIELD$
    118 000000                                  $SBEXP SNUM,,,LS$NUM,+L$FLX            ; DLM number
    119 000000                                  $NEXT
    120 000124                                 FIELD$
    121 000000                                  $SBEXP SOWN,,,LS$OWN,+L$FLG            ; Owner
    122 000000                                  $NEXT
    123 000134                                 FIELD$
    124 000000                                  $SBEXP SCUS,,,LS$CUS,+L$FLX            ; Usage
    125 000000                                  $NEXT
    126 000144                                 FIELD$
    127 000000                                  $SBEXP SDTE,,,LS$DTE,+L$FLX            ; DTE
    128 000000                                  $NEXT
    129 000154                                 FIELD$
    130 000000                                  $SBEXP SCHN,,,LS$CHN,+L$FLX            ; Channel
    131 000000                                  $NEXT
    132 000164                                 FIELD$
    133 000000                                  $SBEXP SCMB,,,LS$CMB,+L$FLX            ; Maximum data
    134 000000                                  $NEXT
    135 000174                                 FIELD$
    136 000000                                  $SBEXP SMXW,,,LS$MXW,+L$FLX            ; Maximum window
```

```
                                                                     I 13
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 8-1
SET CIRCUIT PARSE TABLE
```

```
    635                                           .SBTTL  OWNACT - CIRCUIT OWNER ACTION ROUTINE
    636                                   ;+
    637                                   ; **-OWNACT-CIRCUIT OWNER ACTION ROUTINE
    638                                   ;
    639                                   ; THIS ROUTINE CHECKS THAT THE CIRCUIT OWNER SPECIFIED IS AN LLC
    640                                   ; PROCESS KNOWN TO THE SYSTEM.
    641                                   ;
    642                                   ; INPUTS:
    643                                   ;       .MSGPT  = POINTER TO IMAGE OWNER NAME
    644                                   ;       R4      = ADDRESS OF CONTEXT AREA
    645                                   ;
    646                                   ; OUTPUTS:
    647                                   ;       IF OWNER IS LEGAL,
    648                                   ;           L$OWN(R4) = RAD50 OWNER NAME
    649                                   ;       ELSE, THE PARSE IS TERMINATED.
    650                                   ;
    651                                           .ENABL  LSB
    652 000230                            NODACT:
    653 000230                                  CALL    $SAVAL                  ; SAVE ALL REGISTERS
    654 000234  016701  000000G                 MOV     .MSGPT,R1               ; POINT TO THE OWNER NAME STRING
    655 000240  105721                          TSTB    (R1)+                   ; IS THIS THE EXECUTOR?
    656 000242  001112                          BNE     PVAERR                  ; BR IF NO
    657 000244  105721                          TSTB    (R1)+                   ; ...
    658 000246  001110                          BNE     PVAERR                  ; ...
    659 000250  012702  114224                  MOV     #^RXPT,R2               ; SET TRANSPORT AS THE OWNER
    660 000254  000432                          BR      7$                      ; CONTINUE
    661 000256                            OWNACT:
    662 000256                                  CALL    $SAVAL                  ; SAVE ALL REGISTERS
    663 000262  016701  000000G                 MOV     .MSGPT,R1               ; POINT TO THE OWNER NAME STRING
    664 000266  112105                          MOVB    (R1)+,R5                ; GET LENGTH OF OWNER NAME
    665 000270  003451                          BLE     30$                     ; IF LE, - ERROR
    666 000272  020527  000003                  CMP     R5,#3                   ; IS THE NAME 1-3 CHARACTERS ?
    667 000276  003074                          BGT     PVAERR                  ; IF GT, NO - ERROR
    668                                   ;
    669                                   ; CONVERT NAME TO RAD50
    670                                   ;
    671 000300  060105                          ADD     R1,R5                   ; POINT TO STRING+1
    672 000302  111546                          MOVB    (R5),-(SP)              ; SAVE NEXT CHAR
    673 000304  112715  000077                  MOVB    #'?,(R5)                ; PUT NON RAD50 CHAR INTO BUFFER
    674 000310  010100                          MOV     R1,R0                   ; POINT TO PROCESS NAME
    675 000312                                  SAVRG   <R1>                    ; SAVE R1
    676 000314                                  CALL    $CAT5                   ; CONVERT TO RAD50
    677 000320  103001                          BCC     5$                      ; IF CC, THREE CHARS CONVERTED
    678 000322  005300                          DEC     R0                      ; ELSE, BACK UP POINTER ONE CHARACTER
    679 000324  010102                    5$:   MOV     R1,R2                   ; SAVE CONVERTED NAME
    680 000326                                  RESRG   <R1>                    ; RESTORE R1
    681 000330  112615                          MOVB    (SP)+,(R5)              ; RESTORE NEXT CHAR IN BUFFER
    682 000332  160100                          SUB     R1,R0                   ; CALC LENGTH OF NAME
    683 000334  120061  177777                  CMPB    R0,-1(R1)               ; DID IT ALL GET CONVERTED ?
    684 000340  001053                          BNE     PVAERR                  ; IF NE, NO - ERROR
    685                                   ;
    686                                   ; CHECK THAT OWNER EXISTS, AND IS AN LLC
    687                                   ;
    688 000342                            7$:   SWSTK$  30$                     ;; ENTER SYSTEM STATE
    689 000346                                  CALL    @PDVID                  ;; GET THE PDV INDEX          ; RJK05
    690 000352  103412                          BCS     10$                     ;; IF CS, NO SUCH PROCESS     ;**-1
    691 000354  110264  000102                  MOVB    R2,L$OWN(R4)            ;; SAVE THE OWNER'S PDV INDEX
```

```
                                        .SBTTL  MXWACT - X.25 Circuit maximum window action routine
                                ;+
                                ; *** MXWACT - X.25 Circuit maximum window action routine
                                ;
                                ; INPUTS
                                ;       .MSGPT  - Pointer to maximum window parameter value
                                ;       R4      - Pointer to context area
                                ;
                                ; OUTPUTS
                                ;       L$MXW(R4) Contains maximum window parameter
                                ;
                                ;-

 001054                         MXWACT:
 001054  016701  000000G                MOV     .MSGPT,R1               ; Get pointer to data
 001060  112164  000252                 MOVB    (R1)+,L$MXW(R4)         ; Get data
 001064                                 RETURN
```

```
                                        .SBTTL  TH1ACT - DEAD THRESHOLD ACTION ROUTINE

                                ;+
                                ;
                                ; TH1ACT - DEAD THRESHOLD ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 1 BYTE DEAD THRESHOLD
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$TH1(R4) - DEAD THRESHOLD VALUE
                                ;
                                ;-
001342                          TH1ACT: CALL    CHKDMP                  ; CHECK FOR DMP
001346  110164  000130                  MOVB    R1,L$TH1(R4)            ; SAVE DEAD THRESHOLD VALUE
001352                                  RETURN
```

```
                                                .SBTTL  ALREIN - ALIAS READ INFORMATION INITIALIZE
                                        ;+
                                        ; **-ALREIN-ALIAS READ INFORMATION INITIALIZE ROUTINE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO INITIALIZE FOR THE ALIAS READ INFORMATION
                                        ; OPERATION.
                                        ;
                                        ; INPUTS:
                                        ;       R0 = OPTION (INFORMATION TYPE)
                                        ;       R1 = ID ADDRESS (ALIAS FORMAT)
                                        ;       R2 = LENGTH OF DATA IN INPUT BUFFER
                                        ;       R3 = ADDRESS OF INPUT BUFFER
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = RETURN CODE
                                        ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                        ;       R3 = ADDESS OF OUTPUT BUFFER
                                        ;       DATA SAVED IN CONTEXT AREA
                                        ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                        ;
                                        ; REGISTERS:
                                        ;       R3, R4, R5 ARE PRESERVED
                                        ;-

000000                                  ALREIN::
000000  004567  000000G                         JSR     R5,$SAVRG               ; SAVE R3-R5
000004  010064  000002                          MOV     R0,A$OPT(R4)            ; SAVE OPTION WORD
000010  010364  000006                          MOV     R3,A$BUF(R4)            ; SAVE BUFFER POINTER
                                        ;
                                        ; CHECK LENGTH OF CONTEXT AREA
                                        ;
000014  012700  177773                          MOV     #ME.MPR,R0              ; ASSUME CONTEXT AREA TOO SMALL
000020  026427  177776  000024                  CMP     -2(R4),#A$LEN           ; IS CONTEXT AREA LARGE ENOUGH ?
000026  002466                                  BLT     30$                     ; IF LT, NO - ERROR
                                        ;
                                        ; CHECK OPTIONS FLAG FOR INFORMATION TYPE
                                        ;
000030  012700  177777                          MOV     #ME.FUN,R0              ; ASSUME ILLEGAL OPTION
000034  026427  000002  000040                  CMP     A$OPT(R4),#MO$CHA       ; IS THIS A "SHOW CHARACTERISTICS"
000042  001403                                  BEQ     10$                     ; IF EQ, YES - VALID
000044  005764  000002                          TST     A$OPT(R4)               ; ELSE, IS IT A "SHOW SUMMARY"
000050  001055                                  BNE     30$                     ; IF NE, NO - ERROR
                                        ;
                                        ; CHECK ALIAS FORMAT
                                        ;
000052                                  10$:    CALL    CHKALI                  ; CHECK ALIAS FORMAT
000056  103452                                  BCS     30$                     ; IF CS, ILLEGAL FORMAT
                                        ;
                                        ; CHECK SCOPE, UNLESS ALIAS TYPE WAS "ALL"
                                        ;
000060  126427  000011  177775                  CMPB    A$TYP(R4),#MF$ALL       ; WAS THE ALIAS TYPE "ALL"
000066  001403                                  BEQ     20$                     ; IF EQ, YES - BYPASS SCOPE CHECK
000070                                          CALL    CHKSCO                  ; CHECK THE SCOPE
000074  103443                                  BCS     30$                     ; IF CS, ILLEGAL SCOPE
                                        ;
                                        ; CHECK FOR JUNK AT END OF MESSAGE
```

```
Table of contents

     5-   52    MACRO CALLS AND LOCAL DATA
     6-   81    ALREOP - ALIAS READ INFORMATION OPERATE
     7-  165    FMTNAM - FORMAT A NAME FIELD
     8-  200    FMTSCO - FORMAT SCOPE PARAMETER
     9-  247    FMTDES - FORMAT DESTINATION PARAMETER
    10-  317    FINDA - FIND A MATCHING ALIAS BLOCK
```

```
                                                  I 3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-3
Symbol table

MP$OCO  000776         MP$SGZ  001644         MS$ATR  000011         MT$NLE  000017         M$$FCS= 000000
MP$OHO  000214         MP$SID  000176         MS$AUT  000000         MT$NON  000001         M$$MGE= 000000
MP$ONA  000764         MP$SIN  000310         MS$CLE  000003         MT$NR4  000005         M$$NET= 000000
MP$ONR  002114         MP$SLI  000156         MS$DED  000004         MT$NTY  000060         M$$OVR= 000000
MP$OQL  001453         MP$SLO  000171         MS$DIE  000003         MT$OCT  000060         NM.CLN  003400
MP$OTI  000777         MP$SMX  002202         MS$DUM  000004         MT$PHA  000002         NM.INI  002400
MP$OTY  001022         MP$SND  000500         MS$FAI  000013         MT$POI  000000         NM.OPR  003000
MP$OUS  000777         MP$SNP  006332         MS$GLO  000000         MT$QP2  000010         NM.VR2  002401
MP$OVE  001010         MP$SNU  000542         MS$HOL  000002         MT$ROU  000000         NM.VR3  002402
MP$OWN  004374         MP$SOB  000524         MS$INA  000002         MT$RO4  000004         N$$ACC= 000001
MP$PAR  000036         MP$SPA  000157         MS$LOA  000003         MT$SEC  000000         N$$BUF= 000001
MP$PAS  000513         MP$SPR  000536         MS$LOO  000002         MT$SGD  000020         N$$LDV= 000001
MP$PCH  002152         MP$SPS  000513         MS$OFF  000001         MT$SYS  000002         N$$MCP= 000001
MP$PCO  000024         MP$STA  000000         MS$ON   000000         MT$TER  000001         N$$MLL= 000001
MP$PCT  000144         MP$STI  002140         MS$PRO  000006         MT$TRI  000002         N$$MOV= 000010
MP$PDT  002114         MP$STT  002201         MS$REA  000004         MT$TYP  007777         N$$NCT= 000001
MP$PHA  000170         MP$STY  000175         MS$REF  000001         MT$USD  000000         N$$PEM= 000001
MP$PHY  000012         MP$SUB  000001         MS$RES  000003         MT$X25  000003         OVR     000000R
MP$PLI  000025         MP$SUR  000156         MS$RST  000001         MU$INC  000001         OVRL  = 000060
MP$PLN  002140         MP$SUS  000512         MS$RSX  000002         MU$OUT  000002         P$$P45= 000000
MP$PLO  000012         MP$SVR  000163         MS$RT   000005         MU$PER  000000         P$$WRD= 000000
MP$PMC  004374         MP$SWI  002260         MS$SER  000002         MV$III  000000         Q$$OPT= 000010
MP$PNT  002126         MP$SYP  002424         MS$SHU  000002         MV$IV   000001         R$$DER= 000000
MP$PRI  004411         MP$TFL  000156         MS$SRV  000007         MX$ACT  000020         R$$K11= 000001
MP$PRO  002130         MP$TH1  002206         MS$STA  000000         MX$CIR  000020         R$$SND= 000000
MP$PSS  001763         MP$TH2  002205         MS$SYN  000012         MX$CLN  000020         R$$11M= 000000
MP$PST  001762         MP$TH3  002202         MS$TER  000001         MX$CNM  000006         SF$ANM  000001
MP$RET  001631         MP$TLN  000202         MS$TOP  000003         MX$CON  000006         SF$DNM  000002
MP$RFA  001323         MP$TLO  000172         MS$TRI  000005         MX$DAC  000020         S$$WRG= 000000
MP$RMX  002201         MP$TPA  004375         MS$UNR  000005         MX$DTE  000020         S$$YSZ= 007600
MP$RPA  004374         MP$TRI  002164         MS$VMS  000004         MX$FIL  000034         T$$KMG= 000000
MP$RPR  001606         MP$TST  000144         MS.DON  177600         MX$LGC  000777         T$$MIN= 000000
MP$RRT  002213         MP$TYP  002130         MS.MOR  000002         MX$LIN  000020         V$$CTR= 001000
MP$RST  002212         MP$UCS  004407         MS.PAR  000003         MX$LON  000006         W.BATT  000006
MP$RSV  001754         MP$USR  001750         MS.SUC  000001         MX$NMS  000454         W.BFPD  000014
MP$RSZ  001756         MP$VEC  004410         MT$AR4  000003         MX$NOD  000006         W.BHVR  000004
MP$RTI  001616         MP$VER  004406         MT$ASC  000100         MX$OBJ  000006         W.BLGH  000020
MP$RTM  001611         MP$WDF  002165         MT$BIL  000001         MX$OWN  000040         W.BLPD  000016
MP$RVE  001604         MP$WMX  002177         MT$BYS  000011         MX$PAR  000036         W.BLVR  000002
MP$RVT  000156         MP$XMT  002166         MT$CI   000007         MX$PAS  000010         W.BNPD  000015
MP$SAC  000514         MP$XPF  000242         MT$CLE  000077         MX$RAC  000047         W.BOFF  000012
MP$SAD  000543         MP$XPL  000244         MT$COD  000200         MX$RID  000047         W.BPCB  000000
MP$SCA  000310         MP$XPT  000240         MT$CON  000001         MX$RPS  000047         W.BSIZ  000010
MP$SCO  000144         MP$XXX  177777         MT$DMC  000004         MX$SID  000040         X$$DBT= 000000
MP$SCT  002176         MP$5ST  004406         MT$ETH  000006         MX$SNK  000377         $FMTDV= ****** GX
MP$SDU  000203         MS$ACT  000001         MT$HEX  000040         MX$TYP  000077         $HEADR= ****** GX
MP$SDV  000160         MS$ADU  000010         MT$LPB  000005         MX$UID  000020         $SAVRG= ****** GX
MP$SEH  004401         MS$ALO  000007         MT$MAX  000037         M$$CRB= 000124         $$$SYS= 004374
MP$SER  000144         MS$ASE  000006         MT$MUL  000100         M$$CRX= 000000         $$$SYX= 000000
MP$SET  000000

. ABS.  000104    000   (RW,I,GBL,ABS,OVR)
        000722    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics
```

```
                                                  J 3
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-4
Symbol table
```

```
 000422  016001  000000C                    MOV     <H.LUN+<4*<$TMLUN-1>>>(R0),R1 ;; GET THE DEVICE'S UCB ADDRESS


                                            .ENDC   ; DF R$$MPL

 000426  026101  000002             5$:     CMP     U.RED(R1),R1            ;; IS DEVICE REDIRECTED?
 000432  001403                             BEQ     7$                      ;; IF EQ, NO
 000434  016101  000002                     MOV     U.RED(R1),R1            ;; ELSE, FOLLOW REDIRECTION
 000440  000772                             BR      5$                      ;; ...
 000442  011102                     7$:     MOV     (R1),R2                 ;; POINT TO DCB
 000444  026227  000000G 044524             CMP     D.NAM(R2),#"TI          ;; IS THIS FOR TI:?
 000452  001004                             BNE     8$                      ;; IF NE, NO
 000454  017701  000000G                    MOV     @TKTCB,R1               ;; ELSE, GET OUR TCB ADDRESS    ; RJK02
 000460  016101  000000G                    MOV     T.UCB(R1),R1            ;; AND POINT TO TI:'S UCB
 000464  032761  000004  000010     8$:     BIT     #DV.TTY,U.CW1(R1)       ;; IS IT A TERMINAL ?
 000472  001005                             BNE     10$                     ;; IF NE, YES - SAVE THE ADDRESS
 000474                                     RETC    R0                      ;; ELSE SET THE USER C-BIT
 000506  010164  000020             10$:    MOV     R1,A$SCO(R4)            ;; SAVE THE UCB ADDRESS
 000512                                     RETURN                          ;; RETURN TO USER STATE

 000514  103003                     20$:    BCC     40$                     ; IF CC, RETURN
 000516  012700  177760             30$:    MOV     #ME.PVA,R0              ; SET ILLEGAL PARAMETER VALUE ERROR
 000522  000261                             SEC
 000524                             40$:    RETURN
```

```
     68                                                      .SBTTL  ARREIN - READ AREA STATUS INITIALIZATION
     69                                              ;+
     70                                              ; *** -  ARREIN - READ AREA STATUS INITIALIZATION
     71                                              ;
     72                                              ; INPUTS:
     73                                              ;       R0 = OPTION WORD (STATUS/SUMMARY/CHARACTERISTICS...)
     74                                              ;       R1 = ID BUFFER ADDRESS
     75                                              ;       R2 = LENGTH OF DATA IN INPUT BUFFER (ZERO)
     76                                              ;       R3 = ADDRESS OF INPUT BUFFER
     77                                              ;       R4 = ADDRESS OF CONTEXT AREA
     78                                              ;       $MBLUN ALREADY ASSIGNED TO "NS:" DEVICE
     79                                              ;
     80                                              ; OUTPUTS:
     81                                              ;       C-BIT = SUCCESS/FAILURE
     82                                              ;       R0 = NICE PROTOCOL RETURN CODE
     83                                              ;       R2 = LENGTH OF INFORMATION IN OUTPUT BUFFER
     84                                              ;       R3-R5 = PRESERVED
     85                                              ;
     86                                              ;-
     87
     88 000000                                       ARREIN::
     89 000000  004567  000000G                              JSR     R5,$SAVRG               ; SAVE R3-R5
     90
     91 000004  010064  000010                               MOV     R0,A$OPT(R4)            ; SAVE THE OPTIONS WORD
     92 000010  112764  000077  000006                       MOVB    #MXAREA,A$MAX(R4)       ; ASSUME LIMIT IS MAXIMUM
     93
     94 000016  012700  177773                               MOV     #ME.MPR,R0              ; ASSUME CONTEXT AREA TOO SMALL
     95 000022  026427  177776  000027                       CMP     -2(R4),#A$LENX          ; IS CONTEXT AREA TOO SMALL ?
     96 000030  002002                                       BGE     20$                     ; IF GE, OK
     97 000032  000167  000420                       10$:    JMP     150$                    ; IF LT, YES .. ERROR
     98
     99 000036  012700  177777                       20$:    MOV     #ME.FUN,R0              ; ASSUME ILLEGAL OPTION
    100 000042  032764  177717  000010                       BIT     #^C<MO$COU>,A$OPT(R4)   ; CHECK FOR ILLEGAL OPTIONS
    101 000050  001370                                       BNE     10$                     ; IF NE, ERROR
    102 000052  022764  000060  000010                       CMP     #MO$COU,A$OPT(R4)       ; COUNTERS SPECIFIED?
    103 000060  001764                                       BEQ     10$                     ; BR IF YES - ILLEGAL
    104
    105 000062  005702                                       TST     R2                      ; ANY INFORMATION IN BUFFER ?
    106 000064  001362                                       BNE     10$                     ; IF NE, ERROR
    107
    108 000066  112164  000004                               MOVB    (R1)+,A$NUM(R4)         ; CHECK ID FORMAT
    109 000072  100404                                       BMI     30$                     ; IF MI, PLURAL AREAS
    110 000074  112164  000004                               MOVB    (R1)+,A$NUM(R4)         ; CHECK RANGE IF ENTITY-ID
    111 000100  003025                                       BGT     50$                     ; IF GT, NOT PLURAL AREAS
    112 000102  000416                                       BR      40$                     ; ELSE AREA 0 - NOT VALID
    113                                              ;
    114                                              ; KNOWN, ACTIVE, SIGNIFICANT AREAS
    115                                              ;
    116 000104  005064  000004                       30$:    CLR     A$NUM(R4)               ; START KNOWN SCAN FROM AREA 1
    117 000110  052764  000002  000000                       BIS     #AF.KNO,A$FLG(R4)       ; ALL PLURALS LOOK LIKE KNOWN
    118 000116  124127  177777                               CMPB    -(R1),#MF$KNO           ; IS IT A VALID TYPE ?
    119 000122  001414                                       BEQ     50$                     ; IF EQ, YES
    120 000124  121127  177776                               CMPB    (R1),#MF$ACT            ; IS IT A VALID TYPE ?
    121 000130  001411                                       BEQ     50$                     ; IF EQ, YES
    122 000132  121127  177773                               CMPB    (R1),#MF$SIG            ; IS IT A VALID TYPE ?
    123 000136  001406                                       BEQ     50$                     ; IF EQ, YES
    124 000140  012700  177767                       40$:    MOV     #ME.IID,R0              ; ELSE, INVALID ENTITY ID
```

```
                                        .TITLE  ARREOP - AREA READ INFORMATION OPERATE
                                        .IDENT  /V05.00/
                                        .ENABL  LC
                                        .NLIST  CND

                                ;
                                ; COPYRIGHT (C) 1983, 1984, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ; NETWORK MANAGEMENT - AREA READ INFORMATION OPERATE
                                ;
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  25-MAR-83       Dave Rachlis
                                ;       INITIAL CREATION.
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;


```

```
                                                                 I 7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-2
Symbol table

MD$HEL   000002         ME.DON   177600         MO$ACC   000200         MP$BSD   002203         MP$GDT   002222
MD$KCP   000013         ME.FCO   177762         MO$ADD   000002         MP$BSI   002200         MP$GNM   002223
MD$KDP   000034         ME.FIO   177756         MO$ALA   000004         MP$BSP   006331         MP$GRO   002115
MD$KDZ   000'36         ME.FOP   177763         MO$ALI   000000         MP$BUF   000170         MP$GRP   000541
MD$KL    000040         ME.FOR   177776         MO$CHA   000040         MP$BUP   006324         MP$GTY   002224
MD$KMX   000054         ME.FUN   177777         MO$CIR   000003         MP$BUS   001643         MP$HAD   001757
MD$KMY   000052         ME.GRO   177745         MO$CLE   000100         MP$CAC   001750         MP$HBT   002142
MD$LEN   000050         ME.HAR   177750         MO$COU   000060         MP$CAS   001762         MP$HDD   000162
MD$PCL   000011         ME.IID   177767         MO$DAC   000000         MP$CAT   002210         MP$HTM   001612
MD$QNA   000005         ME.LCO   177766         MO$DEF   000000         MP$CCS   004406         MP$HWA   002210
MD$UNA   000001         ME.LPR   177757         MO$DPR   000000         MP$CHN   002141         MP$IAT   001322
MD$WIT   000002         ME.MPR   177773         MO$ENT   000017         MP$CIR   000144         MP$IDE   000144
MD.ABO   000017         ME.MVE   177771         MO$EVE   000100         MP$CLK   002131         MP$IDP   006327
MD.ACC   000010         ME.OPE   177747         MO$INF   000160         MP$CLN   002126         MP$IHO   000215
MD.AOB   000016         ME.PLO   177751         MO$INS   000002         MP$CLT   002211         MP$INA   002177
MD.BOB   000011         ME.PMI   177743         MO$LIN   000001         MP$CMB   002142         MP$IND   002204
MD.DIA   000007         ME.PNA   177752         MO$LOG   000001         MP$CMK   000537         MP$INI   002201
MD.DOB   000015         ME.PRI   177775         MO$MIR   000031         MP$CMX   002153         MP$ITI   000776
MD.DUM   000002         ME.PTY   177772         MO$NAM   000001         MP$CND   000310         MP$LAA   000231
MD.FAI   000014         ME.PVA   177760         MO$NIC   000023         MP$CNU   001753         MP$LAN   000234
MD.FNA   000001         ME.RES   177761         MO$NOD   000000         MP$COB   000311         MP$LAR   000202
MD.FOB   000006         ME.ROO   177754         MO$OFF   000001         MP$CON   002126         MP$LAS   000012
MD.LOA   000001         ME.SIZ   177774         MO$ON    000000         MP$COS   001604         MP$LBS   000203
MD.LSH   000020         ME.SYS   177746         MO$OPT   000001         MP$CPF   000230         MP$LCO   000226
MD.NOB   000012         ME.UCO   177770         MO$PER   000200         MP$CPL   000232         MP$LCT   000156
MD.NON   177777         MF$ACT   177776         MO$PRO   000002         MP$CPT   000226         MP$LHL   000232
MD.PER   000000         MF$ADD   000000         MO$PRS   000001         MP$CPU   000161         MP$LLE   000227
MD.RES   000004         MF$ADJ   177774         MO$REA   000200         MP$CSZ   001755         MP$LLO   004432
MD.ROB   000005         MF$ALL   177775         MO$SET   000000         MP$CUS   002127         MP$LMB   002152
MD.RSH   000013         MF$BYE   000300         MO$STA   000020         MP$CVA   000540         MP$LMX   002200
MD.SDU   000005         MF$CHA   000023         MO$SUM   000000         MP$DAL   002570         MP$LNA   000144
MD.SLO   000003         MF$DUM   000020         MO$TSK   000000         MP$DCO=  001464         MP$LNO   000233
MD.SNA   000000         MF$EVT   000001         MO$VOL   000000         MP$DDT   002177         MP$LOA   000170
MD.TLO   000004         MF$FUP   000264         MO$ZER   000000         MP$DEL   001131         MP$LOG   004514
MD.UNA   000002         MF$KNO   177777         MP$ACB   000012         MP$DES   000156         MP$LOO   000620
MD.UNR   000003         MF$LOA   000017         MP$ACC   000514         MP$DEV   002114         MP$LTM   001613
MD.UOB   000007         MF$LOO   177775         MP$ACT   004526         MP$DFA   001320         MP$LTY   004533
MD.VOL   000006         MF$REA   000024         MP$ADD   000766         MP$DHO=  001465         MP$LWI   000230
ME$ALI   000010         MF$SIG   177773         MP$ADJ   001440         MP$DIA   000173         MP$MAC   004420
ME$AL2   000003         MF$SPF   000302         MP$ADP   006325         MP$DLB   002571         MP$MAD   001630
ME$ARE   000005         MF$SYS   000026         MP$ADS   004406         MP$DLG   004521         MP$MAP   006323
ME$CIR   000003         MF$TES   000022         MP$ALB   000036         MP$DLI=  001466         MP$MAR   001635
ME$EXA   000000         MF$TRA   000301         MP$AMC   001640         MP$DLT   002200         MP$MAV   006321
ME$EXE   000200         MF$TRI   000021         MP$AMH   001641         MP$DRO   001441         MP$MBN   001636
ME$LIN   000001         MF$ZER   000025         MP$ANB   004420         MP$DST   000454         MP$MBR   001637
ME$LOG   000002         ML$ALL   100000         MP$ASB   000024         MP$DTE   002140         MP$MBU   001642
ME$MOD   000004         ML$CLS   000000         MP$ASC   004432         MP$DTY=  001452         MP$MCB   000156
ME$NOD   000000         ML$CON   000001         MP$AUS   000512         MP$DUA   000207         MP$MCO   001632
ME$NON   177777         ML$EXT   000001         MP$BBT   002165         MP$DUC   000210         MP$MDE   004421
ME$OBJ   000007         ML$FIL   000002         MP$BDF   002164         MP$DUM   000202         MP$MHO   001633
ME$OB2   000004         ML$FIR   000000         MP$BFQ   002121         MP$DUP   002127         MP$MLB   000202
ME$PRO   000005         ML$INT   000000         MP$BLK   001616         MP$DVC   002114         MP$MLK   001306
ME$SYS   000006         ML$KNO   140000         MP$BLO   001452         MP$DWE   001321         MP$MLN   001631
ME.BLO   177744         ML$MON   000003         MP$BMX   002176         MP$ELT   000157         MP$MLP   006333
ME.CON   177753         ML$SYS   000002         MP$BNP   006330         MP$ETY   001605         MP$MRB   002171
ME.CST   177765         ML$TOP   000001         MP$BRT   001620         MP$EVE   000311         MP$MRO   001605
ME.DIS   177755         MN$UNL   000377         MP$BSA   002176         MP$FNC   001752         MP$MRP   006322
```

```
                                                                 J 7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-3
Symbol table
```

```
                                        ; CONTEXT AREA OFFSET DEFINITIONS
                                        ;
      000270                                  .ASECT
            000000                            . = 0
            000020                            L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                        ;
                                        ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                        ; as in the CIRCX$ and LOCCX$ macros.
                                        ;
      000000                            L$NAM:
      000000                            L$DDM:  .BLKW   1               ; DEVICE NAME
      000002                            L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
      000003                            L$UNT:  .BLKB   1               ; UNIT NUMBER
      000004                            L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
            000020                              . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
      000020                            L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
      000040                            L$TRB:  .BLKB   1                       ; TRIBUTARY NUMBER
      000041                            L$PFG:  .BLKB   1                       ; PARSE FLAGS
      000042                            L$SLT:  .BLKW   1                       ; Current SLT/PVC address
      000044                            L$NXT:  .BLKW   1                       ; Pointer to next SLT/PVC address
      000046                            L$SNM:  .BLKW   1                       ; REMAINING NUMBER OF SYSTEM LINES
      000050                            L$TPT:  .BLKW   1                       ; Current tributary pointer (-1 for PSI)
      000052                            L$TNM:  .BLKB   1                       ; REMAINING NUMBER OF TRIBUTARIES
      000053                            L$CTB:  .BLKB   1                       ; CURRENT TRIBUTARY NUMBER
      000054                            L$CHN:  .BLKB   1                       ; Channel number / X.25 port number and
      000055                            L$PDV:  .BLKB   1                       ; ... PDV assigned
      000056                            L$MSG:  .BLKW   1                       ; ERROR MESSAGE STRING POINTER
      000060                            L$BUF:  .BLKW   1                       ; SAVED BUFFER POINTER
      000062                            L$OPT:  .BLKB   1                       ; SAVED OPTIONS BYTE
      000063                            L$TYP:  .BLKB   1                       ; LINE-ID FORMAT TYPE
      000064                            L$FLG:  .BLKW   1                       ; FLAG WORD
      000066                            L$SLEN: .BLKB   1                       ; Significant length of circuit name
      000067                            L$PRO:                                  ; Line protocol
      000067                            L$MTYP: .BLKB   1                       ; Network management circuit type
      000070                            L$TSZ:                                  ; Block size for transport circuits
      000070                            L$LTM:                                  ; Listen timer for transport circuits
      000070                            L$FLX:  .BLKW   1                       ; Flags word for X.25 circuit commands
      000072                            L$SCR:  .BLKW   7                       ; SCRATCH BUFFER
      000110                            L$ADJ:  .BLKW   1                       ; Pointer to adjacency database
      000112                            L$NOD:  .BLKW   1                       ; Adjacent node address
      000114                            L$COU:  .BLKW   1                       ; Count of adjacency entries
      000116                            L$PLB:  .BLKW   1                       ; PLB address
      000120                            L$PAR:  .BLKW   1                       ; PARAMETER TYPE
      000122                            L$SYL:  .BLKW   1                       ; System line number
      000124                            L$LEN:                                  ; LENGTH OF CONTEXT AREA
      000000                                    .PSECT
                                        ;
                                        ; OFFSETS INTO SCRATCH BUFFER (L$SCR) FOR DMC BASE TABLE COUNTERS
                                        ;
      000124                                    .ASECT
            000000                              .=0
      000000                            B$5:    .BLKB   1                       ; BASE TABLE + 5
      000001                            B$6:    .BLKB   1                       ; BASE TABLE + 6
      000002                            B$7:    .BLKB   1                       ; BASE TABLE + 7
      000003                            B$10:   .BLKB   1                       ; BASE TABLE + 10
      000004                            B$11:   .BLKB   1                       ; BASE TABLE + 11
      000005                            B$12:   .BLKB   1                       ; BASE TABLE + 12
```

```
                              ;
001406 005302                         DEC     R2                      ; SEE IF NEW ADDRESS IS CONTIGUOUS
001410 020102                         CMP     R1,R2                   ; IS IT ADJACENT ?
001412 001044                         BNE     40$                     ; IF NE, NO - CAN'T EXTEND THIS ENTRY
001414 020163 000002                  CMP     R1,2(R3)                ; ELSE, DOES NEXT ENTRY INCLUDE THIS ONE?
001420 001475                         BEQ     90$                     ; IF EQ, YES - LEAVE IT ALONE
001422 005343                         DEC     -(R3)                   ; ELSE REDUCE LOW BOUND
001424 000405                         BR      30$                     ; GO INCREASE CARDINALITY AND EXIT
                              ;
                              ; CHECK HIGH BOUND OF CURRENT ENTRY
                              ;
001426 060002                 20$:    ADD     R0,R2                   ; GET HIGH ADDRESS IN SET
001430 020102                         CMP     R1,R2                   ; COMPARE AGAINST HIGH ADDRESS
001432 103470                         BLO     90$                     ; IF LO, ALREADY IN SET
001434 001355                         BNE     10$                     ; IF NE, GREATER BY MORE THAN ONE
001436 005743                         TST     -(R3)                   ; ELSE LEAVE HIGH BOUND ALONE,
001440 005243                 30$:    INC     -(R3)                   ; INCREASE CARDINALITY
                              ;
                              ; CHECK FOR ADJACENT ENTRIES THAT CAN BE COMBINED
                              ;
001442 012302                         MOV     (R3)+,R2                ; GET SIZE OF CURRENT ENTRY..
001444 062302                         ADD     (R3)+,R2                ; ..HIGH ADDR. OF CURRENT, POINT TO NEXT
001446 005302                         DEC     R2                      ; ADJUST HIGH ADDRESS
001450 020263 000002                  CMP     R2,2(R3)                ; DO ENTRIES OVERLAP ?
001454 001405                         BEQ     33$                     ; IF EQ, YES
001456 005202                         INC     R2                      ; SEE IF THEY ABUT
001460 020263 000002                  CMP     R2,2(R3)                ; DO THEY ABUT ?
001464 001053                         BNE     90$                     ; IF NE, NO
001466 005213                         INC     (R3)                    ; NO OVERLAP NODE - LEAVE COUNT ALONE
001470 005313                 33$:    DEC     (R3)                    ; DON'T COUNT OVERLAP NODE TWICE
001472 061363 177774                  ADD     (R3),-4(R3)             ; EXTEND PREVIOUS ENTRY
001476 016323 000004          37$:    MOV     4(R3),(R3)+             ; CLOSE UP THE LIST
001502 020305                         CMP     R3,R5                   ;  ...
001504 103774                         BLO     37$                     ;  ...
001506 162766 000004 000000G          SUB     #4,R$R5(SP)             ; RECLAIM SPACE AT LIST END
001514 042764 000002 000000           BIC     #NF.OVR,N$FLAG(R4)      ; CLEAR OVERFLOW BIT
001522 000434                         BR      90$                     ; DONE
                              ;
                              ; ADD A NEW ENTRY IN THE MIDDLE OF THE LIST
                              ;
001524 024343                 40$:    CMP     -(R3),-(R3)             ; POINT TO LAST ENTRY
001526 010500                         MOV     R5,R0                   ; COPY POINTER
001530 032764 000002 000000           BIT     #NF.OVR,N$FLAG(R4)      ; IS CONTEXT AREA FULL ?
001536 001004                         BNE     50$                     ; IF NE, YES - JUST COPY
001540 022525                         CMP     (R5)+,(R5)+             ; ELSE CLEAR SPACE FOR NEW ENTRY
001542 010566 000000G                 MOV     R5,R$R5(SP)             ; UPDATE SAVED R5
001546 000401                         BR      60$

001550 024040                 50$:    CMP     -(R0),-(R0)             ; DROP LAST ENTRY

001552 020003                 60$:    CMP     R0,R3                   ; EVERYTHING SLID DOWN ?
001554 101412                         BLOS    75$                     ; IF LOS, YES
001556 014045                         MOV     -(R0),-(R5)             ; SLIDE A WORD DOWN
001560 000774                         BR      60$                     ; CONTINUE
                              ;
                              ; ADD A NEW ENTRY AT THE END OF THE LIST
                              ;
```

```
BLNLST     CREATED BY  MACRO  ON 29-JUN-85 AT 12:09     PAGE 7

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

ADJDF$           #5-51        5-74
ALUN$S           #5-52
BIAS             #5-52        10-209       10-234       14-391       14-416
CALL              8-130       8-132        9-162        10-227       10-243       11-267       11-279       12-313       13-340       14-408
                  14-425      15-447       15-454       16-482
CEACC$           #5-53        15-447
CIRCX$           #5-53        5-87
CTRDF$           #5-52        5-68
DHBDF$           #5-53        5-75
ECDDB$           #5-52        5-70
HDRDF$           #5-52        5-71
IMSG$            #5-53
MANDF$           #5-51        5-66
MAP              #5-51        10-206       12-300       14-388
MAPECL           #5-58        12-298
MAPXPT           #5-62        10-203       14-385
NORCX$           #5-53        5-85
PDVDF$           #5-52        5-69
RESMAP           #5-51        8-133
RESRG            #5-51        9-163
RETURN            8-134       9-186        10-244       11-284       12-316       13-368       14-411       14-426       15-457       16-591
RNBDF$           #5-51        5-67
SAVMAP           #5-51        8-131
SAVRG            #5-51        9-161
SERDF$           #5-53        5-76
SOB               14-410
SWSTK$            8-130
XPDDB$           #5-52        5-72
XPTDF$           #5-52        5-73
```

```
**FILE**ID**CHKCIR
```

```
                                                                       I 11
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41      PAGE 2

SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE               REFERENCES

L$MSG     000056            #5-63
L$MTYP    000067            #5-63
L$NAM     000000            #5-63
L$NLEN  = 000020            #5-63        5-63        5-63
L$NOD     000112            #5-63
L$NXT     000044            #5-63
L$OPT     000062            #5-63
L$PAR     000120            #5-63
L$PDV     000055            #5-63
L$PFG     000041            #5-63       *7-95
L$PLB     000116            #5-63
L$PRO     000067            #5-63
L$PVC     000004            #5-63
L$SCN     000020            #5-63
L$SCR     000072            #5-63
L$SLEN    000066            #5-63
L$SLT     000042            #5-63
L$SNM     000046            #5-63
L$SYL     000122            #5-63
L$TNM     000052            #5-63
L$TPT     000050            #5-63
L$TRB     000040            #5-63
L$TSZ     000070            #5-63
L$TYP     000063            #5-63       *7-98
L$UNT     000003            #5-63
ME.IID    177767             7-108
MF$ACT    177776             7-93
MF$SIG    177773             7-91
PRSCIR  = ******   GX        7-104
```

```
                                                                       J 11
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41      PAGE 3

MACRO CROSS REFERENCE                                    CREF   04.00
```

```
                                                                   I 12
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

BIT0     = 000001          #6-63
BIT1     = 000002          #6-63
BIT10    = 002000          #6-63
BIT11    = 004000          #6-63
BIT12    = 010000          #6-63
BIT13    = 020000          #6-63
BIT14    = 040000          #6-63
BIT15    = 100000          #6-63
BIT2     = 000004          #6-63
BIT3     = 000010          #6-63
BIT4     = 000020          #6-63
BIT5     = 000040          #6-63
BIT6     = 000100          #6-63
BIT7     = 000200          #6-63
BIT8     = 000400          #6-63
BIT9     = 001000          #6-63
B$10       000003          #5-63
B$11       000004          #5-63
B$12       000005          #5-63
B$5        000000          #5-63
B$6        000001          #5-63
B$7        000002          #5-63
B$7S10     000006          #5-63
CHKLIN     000000 RG       #7-87
LF$END   = 000020          #6-63
LF$MLT   = 040000          #6-63
LF$REA   = 000001          #6-63
LF$SEG   = 100000          #6-63
LF$SIG   = 000040          #6-63
LF$SKP   = 000004          #6-63
LF$VR2   = 000010          #6-63
LF$ZER   = 000002          #6-63
LP$MPT   = 000010          #6-63
LP$MUX   = 000004          #6-63
LP$TRB   = 000002          #6-63
LP$UNT   = 000001          #6-63        7-95
LP$WCN   = 000040          #6-63        6-63        7-95
LP$WDV   = 000020          #6-63        6-63        7-95
LP$WLD   = 000360          #6-63
LP$WTR   = 000200          #6-63        6-63
LP$WUN   = 000100          #6-63        6-63        7-95
L$ADJ      000110          #5-63
L$BUF      000060          #5-63
L$CHN      000054          #5-63
L$COU      000114          #5-63
L$CTB      000053          #5-63
L$CTL      000002          #5-63
L$DDM      000000          #5-63
L$FLG      000064          #5-63
L$FLX      000070          #5-63
L$LEN      000124          #5-63
L$LTM      000070          #5-63
```

```
                                                                   J 12
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42      PAGE 2

SYMBOL CROSS REFERENCE                                   CREF    04.00
```

J 13
```
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 8-1
SET CIRCUIT PARSE TABLE

    137 000000                                  $NEXT
    138 000204                                 FIELD$
    139 000000                                  $SBEXP STAD,,,LS$TAD,+L$FLG           ; Tributary
    140 000000                                  $NEXT
    141 000214                                 FIELD$
    142 000000                                  $SBEXP SPLT,,,LS$PLT,+L$FLG           ; Poll timer
    143 000000                                  $NEXT
    144 000224                                 FIELD$
    145 000000                                  $SBEXP SBBT,,,LS$BBT,+L$FLG           ; Babble timer
    146 000000                                  $NEXT
    147 000234                                 FIELD$
    148 000000                                  $SBEXP SNMT,,,LS$NMT,+L$FLG           ; Normal timer
    149 000000                                  $NEXT
    150 000244                                 FIELD$
    151 000000                                  $SBEXP SXMT,,,LS$XMT,+L$FLG           ; TX timer
    152 000000                                  $NEXT
    153 000254                                 FIELD$
    154 000000                                  $SBEXP SBSA,,,LS$BSA,+L$FL1           ; Active base
    155 000000                                  $NEXT
    156 000264                                 FIELD$
    157 000000                                  $SBEXP SBSD,,,LS$BSD,+L$FL1           ; Dying base
    158 000000                                  $NEXT
    159 000274                                 FIELD$
    160 000000                                  $SBEXP SBSI,,,LS$BSI,+L$FL1           ; Inactive base
    161 000000                                  $NEXT
    162 000304                                 FIELD$
    163 000000                                  $SBEXP SINA,,,LS$INA,+L$FL1           ; Active increment
    164 000000                                  $NEXT
    165 000314                                 FIELD$
    166 000000                                  $SBEXP SIND,,,LS$IND,+L$FL1           ; Dying increment
    167 000000                                  $NEXT
    168 000324                                 FIELD$
    169 000000                                  $SBEXP SINI,,,LS$INI,+L$FL1           ; Inactive increment
    170 000000                                  $NEXT
    171 000334                                 FIELD$
    172 000000                                  $SBEXP STH1,,,LS$TH1,+L$FL1           ; Dead threshold
    173 000000                                  $NEXT
    174 000344                                 FIELD$
    175 000000                                  $SBEXP STH2,,,LS$TH2,+L$FL1           ; Dying threshold
    176 000000                                  $NEXT
    177 000354                                 FIELD$
    178 000000                                  $SBEXP STH3,,,LS$TH3,+L$FL1           ; Inactive threshold
    179 000000                                  $NEXT
    180 000364                                 FIELD$
    181 000000                                  $SBEXP SMXB,END,,LS$MXB,+L$FL1        ; Max blocks
    182 000000                                  $NEXT
    183 000376                                 FIELD$
    184 000000                                  $SBEXP SACT,,,LS$ACT,+L$FLG           ; Active polling ratio
    185 000000                                  $NEXT
    186 000406                                 FIELD$
    187 000000                                  $SBEXP SLOC,,,LS$NTL,+L$FLG           ; Location
    188 000000                                  $NEXT
    189 000416                                  $NULL  ,END
    190                                 ;
    191                                 ; STATE FOR CLEAR CIRCUIT OPTIONS
    192                                 ;
    193 000000                                 FIELD$  CLECIR
```

J 13
```
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 8-2
SET CIRCUIT PARSE TABLE
```

```
   692 000360  001407                              BEQ     10$                     ;; IF EQ, ILLEGAL (PROCESS = AUX)
   693 000362  067702  000000G                     ADD     @PDVTA,R2               ;; GET THE PDV ADDRESS          ; RJK05
   694 000366  011202                              MOV     (R2),R2                 ;; POINT TO THE PDV             ;**-1
   695 000370  132762  000004  000010              BITB    #ZF.LLC,Z.FLG(R2)       ;; IS THE PROCESS AN LLC ?
   696 000376  001005                              BNE     20$                     ;; IF NE, YES - LEGAL OWNER
   697 000400                              10$:    RETC    R0                      ;; SET USER C-BIT
   698 000412                              20$:    RETURN                          ;; BACK TO USER STATE
   699
   700 000414  103050                      30$:    BCC     PAREND                  ; IF CC, VALID NAME
   701 000416  000424                              BR      PVAERR                  ; ELSE, PARAMETER VALUE ERROR
   702                                             .DSABL  LSB
```

```
    997                                                  .SBTTL  RETACT - X.25 Circuit recall timer action routine
    998                                          ;+
    999                                          ;*** RETACT - X.25 Circuit recall timer action routine
   1000                                          ;
   1001                                          ; INPUTS
   1002                                          ;       .MSGPT  - Pointer to recall timer parameter value
   1003                                          ;       R4      - Pointer to context area
   1004                                          ;
   1005                                          ; OUTPUTS
   1006                                          ;       L$RET(R4) Contains recall timer parameter
   1007                                          ;-
   1008
   1009 001066                                   RETACT:
   1010 001066  016701  000000G                         MOV     .MSGPT,R1               ; Get pointer to data
   1011 001072  112164  000256                          MOVB    (R1)+,L$RET(R4)         ; Get low byte of data
   1012 001076  111164  000257                          MOVB    (R1),L$RET+1(R4)        ; Get high byte of data
   1013 001102                                          RETURN
```

```
                                    .SBTTL  TH2ACT - DYING THRESHOLD ACTION ROUTINE

                              ;+
                              ;
                              ; TH2ACT - DYING THRESHOLD ACTION ROUTINE
                              ;
                              ; INPUTS:
                              ;       .MSGPT - POINTER TO 1 BYTE DYING THRESHOLD
                              ;       R4     - POINTER TO CONTEXT AREA
                              ;
                              ; OUTPUTS:
                              ;       L$TH2(R4) - DYING THRESHOLD VALUE
                              ;
                              ;-
001354                        TH2ACT: CALL    CHKDMP                  ; CHECK FOR DMP
001360  110164  000131                MOVB    R1,L$TH2(R4)            ; SAVE DYING THRESHOLD VALUE
001364                                RETURN
```

```
                                          ;
000076  012700  177776                    20$:    MOV     #ME.FOR,R0              ; ASSUME ILLEGAL INPUT FORMAT
000102  005702                                    TST     R2                      ; ANYTHING LEFT IN INPUT BUFFER ?
000104  001037                                    BNE     30$                     ; IF NE, YES - ILLEGAL INPUT
                                          ;
                                          ; SAVE THE NAMES AND SCOPES OF THE ALIASES REQUESTED FOR LATER LOOK-UP
                                          ;
000106                                            CALL    SAVALI                  ; SAVE THE ALIAS(ES) NAMES AND SCOPES
                                          ;
                                          ; FIGURE OUT IF WE WERE SUCCESSFUL OR NOT
                                          ;
000112  012700  000002                            MOV     #MS.MOR,R0              ; ASSUME SUCCESS, WITH MORE TO FOLLOW
000116  012764  177777  000020                    MOV     #-1,A$SCO(R4)           ; SET UNIQUE VALUE FOR OPERATE ROUTINE
000124  105764  000010                            TSTB    A$NUM(R4)               ; WERE ANY ALIASES FOUND ?
000130  001025                                    BNE     30$                     ; IF NE, YES - RETURN WITH STATUS
000132  012764  000010  000004                    MOV     #ME$ALI,A$PAR(R4)       ; SET ERROR DETAIL-ASSUME NOT V2.0 NCP
000140  016700  000000G                           MOV     CURCTX,R0               ; GET ADDRESS OF CONTEXT AREA
000144  032760  000000G 000000G                   BIT     #CF$VR2,C$FLAG(R0)      ; CONNECTED TO V2.0 NCP?
000152  001403                                    BEQ     25$                     ; BR IF NO
000154  012764  000003  000004                    MOV     #ME$AL2,A$PAR(R4)       ; ELSE SET V2.0 ENTITY ID
000162  012700  177770                    25$:    MOV     #ME.UCO,R0              ; ASSUME INVALID ALIAS-ID (NO MATCHES)
000166  105764  000011                            TSTB    A$TYP(R4)               ; REQUEST FOR SPECIFIC ALIAS ?
000172  003004                                    BGT     30$                     ; IF GT, YES - ERROR (NO ALIAS FOUND)
000174  012700  177600                            MOV     #ME.DON,R0              ; ELSE, NONE TO DISPLAY - DONE !
000200  005064  000004                            CLR     A$PAR(R4)               ; CLEAR ERROR DETAIL
                                          ;
                                          ; USE THE STATUS IN R0 TO BUILD THE RETURN MESSAGE AND C-BIT RETURN
                                          ;
000204  012702  000001                    30$:    MOV     #1,R2                   ; ASSUME 1 BYTE RETURN MESSAGE
000210  016403  000006                            MOV     A$BUF(R4),R3            ; RESTORE THE BUFFER ADDRESS
000214  052764  000001  000000                    BIS     #AF.INI,A$FLG(R4)       ; ASSUME INIT WAS SUCCESSFUL
000222  110013                                    MOVB    R0,(R3)                 ; SET THE RETURN CODE
000224  100013                                    BPL     50$                     ; IF PL, SUCCESSFUL, FINISHED
000226  005064  000000                            CLR     A$FLG(R4)               ; ELSE, UNSUCCESSFUL INIT
000232  116463  000004  000001                    MOVB    A$PAR(R4),1(R3)         ; SET THE PARAMETER TYPE IN ERROR
000240  001404                                    BEQ     40$                     ; IF EQ, THERE WAS NONE
000242  105063  000002                            CLRB    2(R3)                   ; ELSE, MAKE PARAMETER TYPE 2 BYTES LONG
000246  012702  000003                            MOV     #3,R2                   ; SET LENGTH OF RETURN TO 3 BYTES
000252  000261                            40$:    SEC                             ; SET ERROR RETURN
000254                                    50$:    RETURN
```

```
                                          SBTTL   SAVALI - SAVE ALIAS CONTEXT
```

```
        .TITLE  ALREOP - ALIAS READ INFORMATION OPERATE
        .IDENT  /V05.00/
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - ALIAS READ INFORMATION OPERATE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 10-4
Symbol table


Work  file  reads: 208
Work  file writes: 141
Size of work file: 28079 Words  ( 110 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:28.02
SY:ALREOP.V2,[135,134]ALREOP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ALREOP
```

```
ALREOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:06      PAGE 1

SYMBOL CROSS REFERENCE                                   CREF   04.00
```

```
                                        .SBTTL  CHKPAR - CHECK PARAMETER TYPE
                                ;+
                                ; **-CHKPAR-CHECK PARAMETER TYPE
                                ;
                                ; THIS ROUTINE IS CALLED TO CHECK THAT THE NEXT PARAMETER IN THE
                                ; INPUT BUFFER IS OF THE TYPE SPECIFIED.
                                ;
                                ; INPUTS:
                                ;       R1 = PARAMETER TYPE TO CHECK FOR
                                ;       R2 = NUMBER OF BYTES REMAINING IN INPUT BUFFER
                                ;       R3 = ADDRESS OF INPUT BUFFER
                                ;
                                ; OUTPUTS:
                                ;       IF C-CLEAR
                                ;          SUCCESS, PARAMETER TYPE IN R1, R2 AND R3 UPDATED
                                ;       IF C-SET
                                ;          FAILURE, ERROR CODE IN R0
                                ;
                                ; REGISTERS:
                                ;       R1, R4 ARE PRESERVED
                                ;-

000526                          CHKPAR::
000526  112346                          MOVB    (R3)+,-(SP)             ; GET PARAMETER TYPE
000530  112366  000001                  MOVB    (R3)+,1(SP)             ; GET HIGH BYTE
000534  012700  177743                  MOV     #ME.PMI,R0              ; ASSUME PARAMETER MISSING
000540  162702  000002                  SUB     #2,R2                   ; SUB LENGTH OF TYPE FROM BUFF COUNT
000544  002410                          BLT     10$                     ; IF LT, ERROR
000546  021601                          CMP     (SP),R1                 ; IS PARAMETER MISSING?
000550  101006                          BHI     10$                     ; BR IF YES
000552  012700  177772                  MOV     #ME.PTY,R0              ; ASSUME UNKNOWN PARAMETER TYPE
000556  021601                          CMP     (SP),R1                 ; ILLEGAL PARAMETER
000560  001002                          BNE     10$                     ; IF NE, NO - ERROR
000562  000241                          CLC                             ; CLEAR CARRY
000564  000401                          BR      20$                     ; AND RETURN
000566  000261                  10$:    SEC                             ; SET ERROR INDICATOR
000570  005226                  20$:    INC     (SP)+                   ; CLEAN UP STACK
000572                                  RETURN

        000001                          .END
```

```
000144  012764  000005  000016          MOV     #ME$ARE,A$PAR(R4)       ; PARAMETER = AREA
000152  000541                          BR      150$                    ; AND EXIT WITH ERROR
                                ;
                                ; CHECK FOR XPT AND AREA DATABASE. MAKE SURE REQUESTED AREA
                                ; IS KNOWN IN DATABASE. DETRMINE EXEC NODE TYPE.
                                ;
000154                          50$:    SWSTK$  110$                    ;; ENTER SYSTEM STATE

000160  017700  000000G                 MOV     @DECPT,R0               ;; GET DECNET HOME BLOCK ADDRESS; RJK03
000164  016000  000014                  MOV     D$LNUM(R0),R0           ;; EXEC ADDRESS
000170  000300                          SWAB    R0                      ;; GET EXECUTOR AREA
000172  006200                          ASR     R0                      ;;  ...
000174  006200                          ASR     R0                      ;;  ...
000176  042700  177700                  BIC     #^C77,R0                ;; ISOLATE LOW ORDER BITS
000202  010064  000014                  MOV     R0,A$EXA(R4)            ;; SAVE EXECUTOR AREA

000206  012702  114224                  MOV     #^RXPT,R2               ;; ELSE SET UP PROCESS NAME
000212                                  CALL    @PDVID                  ;; GET PDV INDEX             ; RJK03
000216  012766  177765  000000G         MOV     #ME.CST,R$R0(SP)        ;; ASSUME COMPONENT IN WRONG STATE
000224  012764  000000  000016          MOV     #ME$NOD,A$PAR(R4)       ;;  ..(EXECUTOR) NODE
000232  103465                          BCS     90$                     ;; IF CS, NO XPT PROCESS !
000234  067702  000000G                 ADD     @PDVTA,R2               ;; POINT TO PDV ADDRESS      ; RJK03
000240  011202                          MOV     (R2),R2                 ;; POINT TO PDV
000242  005762  000000                  TST     Z.DSP(R2)               ;; PROCESS LOADED ?
000246  001457                          BEQ     90$                     ;; IF EQ, NO !
000250  010264  000012                  MOV     R2,A$XPT(R4)            ;; SAVE PDV POINTER
000254  016202  000016                  MOV     Z.DAT(R2),R2            ;; POINT TO PROCESS DDB

                                        .IF     DF,R$$MPL

                                        SAVMAP                          ;; SAVE CURRENT KISAR6 MAPPING
                                        MOV     @SAHDB,@KSAR6           ;; MAP TO MY EXTERNAL HEADER
                                        MOV     @SAHPT,R0               ;; POINT TO THE HEADER
                                        MOV     H.LUN+<<$MBLUN-1>*4>(R0),R0 ;; POINT TO NS: UCB
                                        RESMAP                          ;; RESTORE PREVIOUS KISAR6

                                        .IFF    ; DF R$$MPL

000260  017700  000000G                 MOV     @HEADR,R0               ;; POINT TO TASK HEADER      ; RJK03
000264  016000  000000C                 MOV     H.LUN+<<$MBLUN-1>*4>(R0),R0 ;; POINT TO NS: UCB

                                        .ENDC   ; DF R$$MPL

000270  032760  000200  000014          BIT     #NF.END,U.CW3(R0)       ;; IS THIS AN ENDNODE ?
000276  001403                          BEQ     60$                     ;; IF EQ, NO
000300  052764  000040  000000          BIS     #AF.END,A$FLG(R4)       ;; ELSE REMEMBER IT
000306  052764  000100  000000  60$:    BIS     #AF.LV2,A$FLG(R4)       ;; ASSUME AREA ROUTER
000314  032760  000100  000014          BIT     #NF.LV2,U.CW3(R0)       ;; IS EXECUTOR AN AREA ROUTER ?
000322  001016                          BNE     80$                     ;; IF NE, YES
000324  042764  000100  000000          BIC     #AF.LV2,A$FLG(R4)       ;; NO
                                ;
                                ; END NODE OR LEVEL 1 ROUTER: ONLY KNOW ABOUT EXECUTOR AREA
                                ;
000332  032764  000002  000000          BIT     #AF.KNO,A$FLG(R4)       ;; PROCESSING PLURAL ENTITIES ?
000340  001406                          BEQ     70$                     ;; IF EQ, NO
000342  016464  000014  000004          MOV     A$EXA(R4),A$NUM(R4)     ;; JUST KNOW ABOUT EXECUTOR AREA
000350  042764  000002  000000          BIC     #AF.KNO,A$FLG(R4)       ;; ONLY RETURN INFO FOR 1 AREA
```

```
     51                                             .SBTTL  MACRO CALLS AND LOCAL DATA
     52                                     ;
     53                                     ; LIBRARY MACRO CALLS
     54                                     ;
     55
     56                                             .MCALL  MANDF$,XPDDB$,ADJDF$,PDVDF$,MBILD$,SLTDF$,DHBDF$,ARRCX$
     57                                             .MCALL  SAVMAP,RESMAP,SAVRG,RESRG,MAP,BIAS,RETC,ASR$
     58
     59 000000                                      ADJDF$                          ; DEFINE ADJACENCY OFFSETS
     60 000000                                      DHBDF$                          ; DEFINE DECNET HOME BLOCK
     61 000000                                      MANDF$                          ; DEFINE NETWORK MANAGEMENT FLAGS
     62 000000                                      MBILD$                          ; DEFINE MESSAGE BUILD SYMBOLS
     63 000000                                      PDVDF$                          ; DEFINE PDV OFFSETS
     64 000000                                      SLTDF$                          ; DEFINE SLT OFFSETS
     65 000000                                      XPDDB$                          ; DEFINE XPT DATABASE OFFSETS
     66
     67 000000                                      ARRCX$  LIST                    ; DEFINE CONTEXT AREA
                                            ;
                                            ; CONTEXT AREA DEFINITION
                                            ;
        000004                                      .ASECT
                     000000                         . = 0
        000000                              A$FLG:  .BLKW   1                       ; FLAGS WORD
        000002                              A$BUF:  .BLKW   1                       ; OUTPUT BUFFER ADDRESS
        000004                              A$NUM:  .BLKW   1                       ; AREA NUMBER
        000006                              A$MAX:  .BLKW   1                       ; HIGHEST AREA IN DATABASE
        000010                              A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
        000012                              A$XPT:  .BLKW   1                       ; XPT PDV ADDRESS
        000014                              A$EXA:  .BLKW   1                       ; EXECUTOR AREA
        000016                              A$PAR:  .BLKW   1                       ; PARAMETER FOR RESPONSE MESSAGE
        000020                              A$NAM:  .BLKB   7                       ; NODENAME BUFFER
        000027                              A$LENX:
        000000                                      .PSECT
                                            ;
                                            ; FLAGS WORD BIT DEFINITIONS
                                            ;
                     000001                         AF.INI  =       1               ; INITIALIZATION FLAG
                     000002                         AF.KNO  =       2               ; KNOWN/ACTIVE/SIGNIFIGANT AREAS
                     000020                         AF.MUX  =       20              ; CIRCUIT IS MUX
                     000040                         AF.END  =       40              ; EXECUTOR IS AN ENDNODE
                     000100                         AF.LV2  =       100             ; EXECUTOR IS AN AREA ROUTER
                     000077                         MXAREA  =       63.             ; MAXIMUM AREA NUMBER
     68
     69                                     ;
     70                                     ; LOCAL DATA
     71                                     ;
     72 000000                              TEMP:   .BLKW   1                       ; TEMPORARY
     73
```

```
                                                                       J  7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-3
Symbol table

MP$MRT  002153          MP$RMX  002201          MP$VER  004406          MT$ETH  000006          M$$NET= 000000
MP$MRV  006320          MP$RPA  004374          MP$WDF  002165          MT$HEX  000040          M$$OVR= 000000
MP$MSB  000170          MP$RPR  001606          MP$WMX  002177          MT$LPB  000005          NM.CLN  003400
MP$MVE  000145          MP$RRT  002213          MP$XMT  002166          MT$MAX  000037          NM.INI  002400
MP$MVI  001634          MP$RST  002212          MP$XPF  000242          MT$MUL  000100          NM.OPR  003000
MP$MVR  001751          MP$RSV  001754          MP$XPL  000244          MT$NLE  000017          NM.VR2  002401
MP$MWN  002154          MP$RSZ  001756          MP$XPT  000240          MT$NON  000001          NM.VR3  002402
MP$MXB  002172          MP$RTI  001616          MP$XXX  177777          MT$NR4  000005          NODNAM  001416R
MP$MXC  000466          MP$RTM  001611          MP$5ST  004406          MT$NTY  000060          NXTNOD  001350R
MP$MXR  001630          MP$RVE  001604          MS$ACT  000001          MT$OCT  000060          N$ADJ1  000072
MP$MXW  002143          MP$RVT  000156          MS$ADU  000010          MT$PHA  000002          N$ADJ2  000074
MP$NAA  006335          MP$SAC  000514          MS$ALO  000007          MT$POI  000000          N$CACH  000062
MP$NAC  001130          MP$SAD  000543          MS$ASE  000006          MT$QP2  000010          N$CRC   000120
MP$NAP  006326          MP$SCA  000310          MS$ATR  000011          MT$ROU  000000          N$HC1   000052
MP$NCT  000240          MP$SCO  000144          MS$AUT  000000          MT$RO4  000004          N$HC2   000056
MP$NET  002114          MP$SCT  002176          MS$CLE  000003          MT$SEC  000000          N$LV1   000002
MP$NLI  000765          MP$SDU  000203          MS$DED  000004          MT$SGD  000020          N$LV2   000010
MP$NNA  000764          MP$SDV  000160          MS$DIE  000003          MT$SYS  000002          N$MHC1  000036
MP$NOD  000500          MP$SEH  004401          MS$DUM  000004          MT$TER  000001          N$MHC2  000044
MP$NRA  006334          MP$SER  000144          MS$FAI  000013          MT$TRI  000002          N$PLD   000016
MP$NRB  000214          MP$SET  000000          MS$GLO  000000          MT$TYP  007777          N$PRI   000076
MP$NSA  001617          MP$SGZ  001644          MS$HOL  000002          MT$USD  000000          N$ROA1  000022
MP$NTI  002141          MP$SID  000176          MS$INA  000002          MT$X25  000003          N$ROA2  000030
MP$NUM  001642          MP$SIN  000310          MS$LOA  000003          MU$INC  000001          N$RTMX  000014
MP$NVE  001274          MP$SLI  000156          MS$LOO  000002          MU$OUT  000002          N$RTM1  000014
MP$NXN= 001476          MP$SLO  000171          MS$OFF  000001          MU$PER  000000          N$RTM2  000015
MP$OAC  000632          MP$SMX  002202          MS$ON   000000          MV$III  000000          N$RT1   000000
MP$OAN  000620          MP$SND  000500          MS$PRO  000006          MV$IV   000001          N$RT2   000006
MP$OCO  000776          MP$SNP  006332          MS$REA  000004          MXAREA= 000077          N$TCTL  000112
MP$OHO  000214          MP$SNU  000542          MS$REF  000001          MX$ACT  000020          N$TLC   000100
MP$ONA  000764          MP$SOB  000524          MS$RES  000003          MX$CIR  000020          N$TNC   000114
MP$ONR  002114          MP$SPA  000157          MS$RST  000001          MX$CLN  000020          N$TRC   000106
MP$OQL  001453          MP$SPR  000536          MS$RSX  000002          MX$CNM  000006          N$TTCB  000110
MP$OTI  000777          MP$SPS  000513          MS$RT   000005          MX$CON  000006          N$VER   000066
MP$OTY  001022          MP$STA  000000          MS$SER  000002          MX$DAC  000020          N$XLEN  000124
MP$OUS  000777          MP$STI  002140          MS$SHU  000002          MX$DTE  000020          N$$ACC= 000001
MP$OVE  001010          MP$STT  002201          MS$SRV  000007          MX$FIL  000034          N$$BUF= 000001
MP$OWN  004374          MP$STY  000175          MS$STA  000000          MX$LGC  000777          N$$LDV= 000001
MP$PAR  000036          MP$SUB  000001          MS$SYN  000012          MX$LIN  000020          N$$MCP= 000001
MP$PAS  000513          MP$SUR  000156          MS$TER  000001          MX$LON  000006          N$$MLL= 000001
MP$PCH  002152          MP$SUS  000512          MS$TOP  000003          MX$NMS  000454          N$$MOV= 000010
MP$PCO  000024          MP$SVR  000163          MS$TRI  000005          MX$NOD  000006          N$$NCT= 000001
MP$PCT  000144          MP$SWI  002260          MS$UNR  000005          MX$OBJ  000006          N$$PEM= 000001
MP$PDT  002114          MP$SYP  002424          MS$VMS  000004          MX$OWN  000040          PDVTA = ****** GX
MP$PHA  000170          MP$TFL  000156          MS.DON  177600          MX$PAR  000036          P$$P45= 000000
MP$PHY  000012          MP$TH1  002206          MS.MOR  000002          MX$PAS  000010          P$$WRD= 000000
MP$PLI  000025          MP$TH2  002205          MS.PAR  000003          MX$RAC  000047          Q$$OPT= 000010
MP$PLN  002140          MP$TH3  002202          MS.SUC  000001          MX$RID  000047          R$R1   = ****** GX
MP$PLO  000012          MP$TLN  000202          MT$AR4  000003          MX$RPS  000047          R$R2   = ****** GX
MP$PMC  004374          MP$TLO  000172          MT$ASC  000100          MX$SID  000040          R$$DER= 000000
MP$PNT  002126          MP$TPA  004375          MT$BIL  000001          MX$SNK  000377          R$$K11= 000001
MP$PRI  004411          MP$TRI  002164          MT$BYS  000011          MX$TYP  000077          R$$SND= 000000
MP$PRO  002130          MP$TST  000144          MT$CI   000007          MX$UID  000020          R$$11M= 000000
MP$PSS  001763          MP$TYP  002130          MT$CLE  000077          M$$CRB= 000124          SF$ANM  000001
MP$PST  001762          MP$UCS  004407          MT$COD  000200          M$$CRX= 000000          SF$DNM  000002
MP$RET  001631          MP$USR  001750          MT$CON  000001          M$$FCS= 000000          SF.ACT= 000200
MP$RFA  001323          MP$VEC  004410          MT$DMC  000004          M$$MGE= 000000          SF.ENA= 000100
```

```
                                                                       K  7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-4
Symbol table
```

```
        000006                          B$7S10: .BLKB   1                       ; SUM OF BASE TABLE 7 AND 10
                                                .EVEN
        000000                                  .PSECT
```

```
    573 001562  032764  000002  000000  70$:    BIT     #NF.OVR,N$FLAG(R4)      ; IS CONTEXT AREA FULL ?
    574 001570  001011                          BNE     90$                     ; IF NE, YES - FORGET NODE
    575 001572  010503                          MOV     R5,R3                   ; ELSE POINT TO FREE ELEMENT
    576 001574  022525                          CMP     (R5)+,(R5)+             ; NEW FREE POINTER
    577 001576  010566  000000G                 MOV     R5,R$R5(SP)             ; SAVE IT
    578
    579 001602  016605  000000G         75$:    MOV     R$R5(SP),R5             ; RETRIEVE POINTER TO LAST ENTRY
    580 001606  012723  000001                  MOV     #1,(R3)+                ; ONLY ONE ELEMENT NOW
    581 001612  010123                          MOV     R1,(R3)+                ; MEMBER IS CURRENT NODE ADDRESS
    582
    583 001614  010503                  90$:    MOV     R5,R3                   ; GET POINTER TO NEXT FREE ENTRY
    584 001616  014364  000050                  MOV     -(R3),N$HCCX(R4)        ; PUT ADDRESS IN HIGHEST CURRENT CONTEXT
    585 001622  064364  000050                  ADD     -(R3),N$HCCX(R4)        ; ADD NUMBER OF NODE ENTRIES
    586 001626  005364  000050                  DEC     N$HCCX(R4)              ; ADJUST ADDRESS
    587 001632  032764  000002  000000          BIT     #NF.OVR,N$FLAG(R4)      ; IS CONTEXT AREA FULL ?
    588 001640  001001                          BNE     100$                    ; IF NE, YES - SET CARRY
    589 001642  005727                          TST     (PC)+                   ; CLEAR CARRY - CONTEXT AREA NOT FULL
    590 001644  000261                  100$:   SEC                             ; SET CARRY - CONTEXT AREA FULL
    591 001646                                  RETURN
    592
    593         000001                          .END
```

```
**FILE**ID**CHKCIR




  CCCCCCCC   HH      HH   KK      KK     CCCCCCCC     IIIIII     RRRRRRRR
  CCCCCCCC   HH      HH   KK      KK     CCCCCCCC     IIIIII     RRRRRRRR
CC           HH      HH   KK      KK   CC               II       RR      RR
CC           HH      HH   KK      KK   CC               II       RR      RR
CC           HH      HH   KK    KK     CC               II       RR      RR
CC           HH      HH   KK    KK     CC               II       RR      RR
CC           HHHHHHHHHH   KKKKKK       CC               II       RRRRRRRR
CC           HHHHHHHHHH   KKKKKK       CC               II       RRRRRRRR
CC           HH      HH   KK    KK     CC               II       RR  RR
CC           HH      HH   KK    KK     CC               II       RR  RR
CC           HH      HH   KK      KK   CC               II       RR    RR         ....
CC           HH      HH   KK      KK   CC               II       RR    RR         ....
  CCCCCCCC   HH      HH   KK      KK     CCCCCCCC     IIIIII     RR      RR       ....
  CCCCCCCC   HH      HH   KK      KK     CCCCCCCC     IIIIII     RR      RR       ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
CHKCIR - CHECK A CIRCUIT-ID FOR MACRO V05.03b Saturday 29-Jun-85 17:41
Table of contents
```

```
CHKCIR     CREATED BY  MACRO  ON 29-JUN-85 AT 17:41     PAGE 3
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

CALL              7-104
CIRCX$           #5-59        5-63
MANDF$           #5-59        5-61
RETURN            7-110
```

```
**FILE**ID**CHKLIN
```

```
                                                                    J 12
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE              REFERENCES

L$MSG      000056           #5-63
L$MTYP     000067           #5-63
L$NAM      000000           #5-63
L$NLEN   = 000020           #5-63         5-63         5-63
L$NOD      000112           #5-63
L$NXT      000044           #5-63
L$OPT      000062           #5-63
L$PAR      000120           #5-63
L$PDV      000055           #5-63
L$PFG      000041           #5-63        *7-95
L$PLB      000116           #5-63
L$PRO      000067           #5-63
L$PVC      000004           #5-63
L$SCN      000020           #5-63
L$SCR      000072           #5-63
L$SLEN     000066           #5-63
L$SLT      000042           #5-63
L$SNM      000046           #5-63
L$SYL      000122           #5-63
L$TNM      000052           #5-63
L$TPT      000050           #5-63
L$TRB      000040           #5-63
L$TSZ      000070           #5-63
L$TYP      000063           #5-63        *7-97
L$UNT      000003           #5-63
ME.IID     177767            7-107
MF$ACT     177776            7-93
MF$SIG     177773            7-91
PRSLIN   = ******    GX      7-103
```

```
                                                                    K 12
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42     PAGE 3
MACRO CROSS REFERENCE                                   CREF    04.00
```

```
    194 000000                                          $SBEXP CALL,END                         ; All
    195 000000                                          $NEXT
    196 000430                                         FIELD$
    197 000000                                          $SBEXP CLCT,,,LS$LCT,+L$FLX             ; Counter timer
    198 000000                                          $NEXT
    199 000440                                         FIELD$
    200 000000                                          $SBEXP CRET,,,LS$RET,+L$FLX             ; Recall timer
    201 000000                                          $NEXT
    202                                          .IFTF
    203                                        ;
    204                                        ; END OF MESSAGE
    205                                        ;
    206 000450                                         FIELD$  END
    207 000000                                          $EOM    ,$EXIT
    208 000000                                          $ERROR ME.GRO
```

```
                                                .SBTTL  TADACT - TRIBUTARY ADDRESS ACTION ROUTINE
                                        ;+
                                        ; **-TADACT-TRIBUTARY ADDRESS ACTION ROUTINE
                                        ;
                                        ; THIS ROUTINE SAVES THE TRIBUTARY ADDRESS IN THE CONTEXT AREA.
                                        ;
                                        ; INPUTS:
                                        ;       .MSGNM  = TRIBUTARY ADDRESS
                                        ;       R4      = ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       L$TAD(R4) = TRIBUTARY ADDRESS
                                        ;-

000420                                  TADACT:
000420  132764  000360  000041                  BITB    #LP$WLD,L$PFG(R4)       ; ANY WILD CARDS ?
000426  001015                                  BNE     PNAERR                  ; IF NE, YES - BAD IDEA !
000430  116701  000000G                         MOVB    .MSGNM,R1               ; GET THE TRIBUTARY ADDRESS
000434  001415                                  BEQ     PVAERR                  ; IF EQ, PARAMETER VALUE ERROR
000436  026427  000000  062204                  CMP     L$DDM(R4),#^RPCL        ; IS THE LINE A PCL ?
000444  001003                                  BNE     10$                     ; IF NE, NO - PROCEDE
000446  020127  000037                          CMP     R1,#31.                 ; IS THE ADDRESS VALID FOR A PCL ?
000452  101006                                  BHI     PVAERR                  ; IF HI, NO - PARAMETER VALUE ERROR
000454  110164  000103                  10$:    MOVB    R1,L$TAD(R4)            ; ELSE, SAVE THE ADDRESS
000460  000426                                  BR      PAREND                  ; END PARAMETER PARSE

                                          .IFTF ; .IF NDF R$$11S
```

```
  1015                                               .SBTTL  PCKBCD - Pack string of digits in BCD format
  1016                                       ;+
  1017                                       ; *** PCKBCD - Pack a string of digits in BCD format
  1018                                       ;
  1019                                       ; INPUTS:
  1020                                       ;       R0 - Number of bytes in packed address
  1021                                       ;       R1 - Address of buffer containing ASCII digits to pack
  1022                                       ;       R2 - Number of digits specified
  1023                                       ;       R5 - Address of buffer to store packed data
  1024                                       ;
  1025                                       ; OUTPUTS:
  1026                                       ;       Buffer specified in R5 contains data in BCD format
  1027                                       ;       All registers destroyed
  1028                                       ;
  1029                                       ;-
  1030
  1031 001104                                PCKBCD:
  1032 001104  010503                                MOV     R5,R3                   ; Get address of packed data
  1033 001106  105023                        10$:    CLRB    (R3)+                   ; Initialize packed address
  1034 001110  005300                                DEC     R0                      ; More to initialize?
  1035 001112  003375                                BGT     10$                     ; BR if yes
  1036 001114  005003                                CLR     R3                      ; Position inidcator (0=high four bits)
  1037 001116  005305                                DEC     R5                      ; Update address for first time
  1038 001120  112100                        20$:    MOVB    (R1)+,R0                ; Get digit to pack
  1039 001122  020027  000060                        CMP     R0,#60                  ; Is this a valid digit?
  1040 001126  103422                                BLO     70$                     ; BR if no (sets carry)
  1041 001130  022700  000071                        CMP     #71,R0                  ; Still valid?
  1042 001134  103417                                BLO     70$                     ; BR if no
  1043 001136  162700  000060                        SUB     #60,R0                  ; Convert to binary
  1044 001142  005703                        40$:    TST     R3                      ; Store in high four bits?
  1045 001144  001007                                BNE     50$                     ; BR if no
  1046 001146                                        ASL$    4,R0                    ; Move to high four bits
  1047 001156  005203                                INC     R3                      ; Indicate store in low four bits next
  1048 001160  005205                                INC     R5                      ; Point to next byte
  1049 001162  000401                                BR      60$                     ; BR to store decimal
  1050 001164  005303                        50$:    DEC     R3                      ; Indicate store in high four bits next
  1051 001166  150015                        60$:    BISB    R0,(R5)                 ; Pack the number into the buffer
  1052 001170  005302                                DEC     R2                      ; More to convert?
  1053 001172  003352                                BGT     20$                     ; BR if yes
  1054 001174                                70$:    RETURN
```

```
                                        .SBTTL  TH3ACT - INACTIVE THRESHOLD ACTION ROUTINE

                                ;+
                                ;
                                ; TH3ACT - INACTIVE THRESHOLD ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 1 BYTE INACTIVE THRESHOLD
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$TH3(R4) - INACTIVE THRESHOLD VALUE
                                ;
                                ;-
 001366                         TH3ACT: CALL    CHKDMP                  ; CHECK FOR DMP
 001372  110164  000132                 MOVB    R1,L$TH3(R4)            ; SAVE INACTIVE THRESHOLD VALUE
 001376                                 RETURN
```

```
                                                  .SBTTL  SAVALI - SAVE ALIAS CONTEXT
                                          ;+
                                          ; **-SAVALI-SAVE ALIAS CONTEXT
                                          ;
                                          ; THIS ROUTINE IS CALLED TO SAVE THE NAME AND SCOPE OF ALL ALIASES
                                          ; TO BE DISPLAYED. THIS CONTEXT IS USED BY THE OPERATE CALL TO
                                          ; LOOK UP EACH ALIAS.
                                          ;
                                          ; INPUTS:
                                          ;       R4 = ADDRESS OF CONTEXT BLOCK WITH THE FOLLOWING OFFSETS:
                                          ;             A$TYP = ALIAS NAME FORMAT TYPE:
                                          ;                       -3 = ALL (MF$ALL)
                                          ;                       -2 = ACTIVE (MF$ACT)
                                          ;                       -1 = KNOWN (MF$KNO)
                                          ;                       >0 = SPECIFIC ALIAS
                                          ;             A$ALI = ALIAS NAME, SPACE FILLED IF A$TYP > 0
                                          ;             A$SCO = SCOPE OF ALIAS(ES)
                                          ;
                                          ; OUTPUTS:
                                          ;       R4 = ADDRESS OF CONTEXT BLOCK WITH THE FOLLOWING OFFSETS:
                                          ;             A$NUM = NUMBER OF ALIAS CONTEXTS SAVED
                                          ;             A$NEX = POINTER TO FIRST ALIAS CONTEXT
                                          ;             A$FLG = AF.OVR SET IF ALIASES OVERFLOWED CONTEXT AREA
                                          ;
                                          ; REGISTERS:
                                          ;       R4 IS PRESERVED.
                                          ;-

000256                                    SAVALI:
000256  105064  000010                            CLRB    A$NUM(R4)               ; SET SAVED COUNT TO 0
000262  010405                                    MOV     R4,R5                   ; COPY POINTER TO CONTEXT BLOCK
000264  062705  000024                            ADD     #A$SAV,R5               ; POINT TO ALIAS CONTEXT SAVE AREA
000270  010564  000022                            MOV     R5,A$NEX(R4)            ; SAVE POINTER TO "NEXT" ALIAS FOR SHOW

000274                                            SWSTK$  90$                     ;; ENTER SYSTEM STATE
000300                                            SAVMAP                          ;; SAVE CURRENT MAPPING
000304  017700  000000G                           MOV     @DECPT,R0               ;; POINT TO DEC HOME BLOCK       ; RJK03
000310  062700  000000                            ADD     #D$ANN,R0               ;; POINT TO ALIAS BLOCK LISTHEAD;**-1
000314  105764  000011                            TSTB    A$TYP (R4)              ;; LOOKING FOR A NAMED ALIAS ?
000320  002406                                    BLT     20$                     ;; IF LT, NO (ALL OR KNOWN)
                                          ;
                                          ; SAVE A SPECIFIC ALIAS CONTEXT
                                          ;
000322                                            CALL    FINDA                   ;; LOOK FOR AN ALIAS NAME BLOCK
000326  103424                                    BCS     80$                     ;; IF CS, DIDN'T EXIST - RETURN
000330                                            CALL    SAVCTX                  ;; SAVE ALIAS CONTEXT
000334  000421                                    BR      80$                     ;; AND RETURN
                                          ;
                                          ; SAVE MULTIPLE ALIASES (KNOWN, ACTIVE OR ALL)
                                          ;
000336  010001                            20$:    MOV     R0,R1                   ;; PUT LISTHEAD ADDRESS IN R1
000340  011146                            30$:    MOV     (R1),-(SP)              ;; SET UP TO GET NEXT ALIAS NAME BLOCK
000342                                            CALL    @CEACC                  ;; MAP TO IT                     ; RJK03
000346  012601                                    MOV     (SP)+,R1                ;; GET MAPPED VIRTUAL ADDRESS OF;**-1
000350  001413                                    BEQ     80$                     ;; IF EQ, FINISHED
000352  126427  000011  177775                    CMPB    A$TYP(R4),#MF$ALL       ;; IS THIS A SHOW ALL OPERATION ?
000360  001404                                    BEQ     40$                     ;; IF EQ, YES - SKIP SCOPE CHECK
```

```
     52                                         .SBTTL  MACRO CALLS AND LOCAL DATA
     53                                 ;
     54                                 ; MACRO LIBRARY CALLS
     55                                 ;
     56                                         .MCALL  ANBDF$,MANDF$,SWSTK$,SAVMAP,MAP,RESMAP,RETC
     57                                         .MCALL  HDRDF$,DHBDF$,ALRCX$
     58
     59 000000                                  ANBDF$                          ; DEFINE ALIAS NAME BLOCK OFFSETS
     60 000000                                  MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     61 000000                                  HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     62 000000                                  DHBDF$                          ; DEFINE DEC HOME BLOCK
     63
     64 000000                                  ALRCX$  LIST                    ; DEFINE CONTEXT AREA
                                        ;
                                        ; CONTEXT AREA OFFSET DEFINITIONS
                                        ;
                000020                          MAXID=16.                       ; MAXIMUM LENGTH OF USER ID
                000010                          MAXPAS=8.                       ; MAXIMUM LENGTH OF PASSWORD
                000020                          MAXACC=16.                      ; MAXIMUM LENGTH OF ACCOUNT
        000020                                  .ASECT
                000000                          .=0
                                        ;
                                        ; OFFSETS IN COMMON WITH CHANGE INFORMATION MODULES
                                        ;
        000000                          A$FLG:  .BLKW   1                       ; FLAG WORD
        000002                          A$OPT:  .BLKW   1                       ; SAVED OPTIONS WORD
        000004                          A$PAR:  .BLKW   1                       ; CURRENT PARAMETER TYPE
        000006                          A$BUF:  .BLKW   1                       ; SAVED POINTER TO BUFFER
        000010                          A$NUM:  .BLKB   1                       ; NUMBER OF BYTES IN ANB (FOR SET)
        000011                          A$TYP:  .BLKB   1                       ; ALIAS FORMAT TYPE
        000012                          A$ALI:  .BLKW   3                       ; ALIAS NAME, SPACE FILLED
        000020                          A$SCO:  .BLKW   1                       ; SCOPE OF ALIAS
                                        ;
                                        ; OFFSETS UNIQUE TO READ INFORMATION PARAMETERS MODULES
                                        ;
        000022                          A$NEX:  .BLKW   1                       ; POINTER TO NEXT ALIAS TO BE DISPLAYED
        000024                          A$LEN:                                  ; LENGTH OF CONTEXT AREA
        000024                          A$SAV:                                  ; SAVE AREA FOR NAMES
        000000                                  .PSECT
                                        ;
                                        ; FLAGS WORD BIT DEFINITIONS
                                        ;
                000001                          AF.INI=1                        ; SUCCESSFUL INITIALIZATION COMPLETED
                000002                          AF.OVR=2                        ; CONTEXT AREA OVERFLOWED
     65
     66                                         .IF NDF R$$PRO  ; PRO/DECNET
     67
     68                                 ;
     69                                 ; ERROR MESSAGE TEXT STRINGS
     70                                 ;
     71                                         .ENABL  LC
     72                                         .NLIST  BEX
     73 000000     111     156     163  OVR:    .ASCII  /Insufficient buffer space for additional ALIASES/
     74         000060                          OVRL=.-OVR
     75
     76                                         .ENDC   ; NDF R$$PRO
     77
```

```
                                                                         K  3
ALREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:06      PAGE 1

SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE               REFERENCES

AF.INI  = 000001            #5-64       6-106
AF.OVR  = 000002            #5-64       6-113
ALREOP    000060 RG         #6-102
A$ALI     000012            #5-64      *6-139      *6-140      *6-141      10-345      10-347      10-349
A$BUF     000006            #5-64      *6-104       6-150       6-156
A$FLG     000000            #5-64       6-106       6-113      *6-161
A$LEN     000024            #5-64
A$NE      000022            #5-64       6-138      *6-144
A$NUM     000010            #5-64      *6-111
A$OPT     000002            #5-64
A$PAR     000004            #5-64
A$SAV     000024            #5-64
A$SCO     000020            #5-64       6-143      *6-146       10-351
A$TYP     000011            #5-64
A.ACC     000020            #5-59
A.LNK     000000            #5-59
A.NAM     000002            #5-59       10-345      10-347      10-349
A.PUD     000010            #5-59
A.REM     000012            #5-59       9-288
A.UCB     000010            #5-59       10-351
CEACC   = ******   GX        10-341
DECPT   = ******   GX        9-279
D$ANN     000000             9-280
FINDA     000640 R           9-281     #10-339
FMTDES    000446 R           6-148     #9-269
FMTNAM    000306 R           6-142     #7-184       9-289
FMTSCO    000354 R           6-147     #8-217
KISAR6  = ******   GX        9-278     *9-302
MAXACC  = 000020            #5-64
MAXID   = 000020            #5-64
MAXPAS  = 000010            #5-64
ME.DON    177600             6-105
ME.OPE    177747             6-115
MP$DES    000156             9-270
MP$SCO    000144             8-218
MS.SUC    000001             6-134
N$$VCT  = ******             9-278      9-283       9-302
OVR       000000 R          #5-73       5-74        6-123
OVRL    = 000060            #5-74       6-121
R$$PRO  = ******             5-66       6-119
$FMTDV  = ******   GX        8-239
$HEADR  = ******   GX        9-283
$SAVRG  = ******   GX        6-103
```

```
                                                                         L  3
ALREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:06      PAGE 2

MACRO CROSS REFERENCE                                      CREF   04.00
```

```
                                                                      K 4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-1
Symbol table

AF.INI= 000001                   D$$YNC= 000000                   MB$DIS   000001                  MC$NPF   001616                  MD$DMF   000046
AF.DVR= 000002                   D$$YNM= 000000                   MB$ENA   000000                  MC$NPR   001630                  MD$DMP   000022
ALIIDX  000026R                  D.NAM = ****** GX                MB$FUL   000002                  MC$NRE   001200                  MD$DMR   000050
ALIT1   000000R                  E$$XPR= 000000                   MB$MIX   000002                  MC$NRT   001166                  MD$DMV   000042
ALIT2   000010R                  F$$LVL= 000001                   MB$ONE   000001                  MC$NVR   001642                  MD$DN    000016
ALIT3   000016R                  G$$TPP= 000000                   MB$RXO   000001                  MC$P08   000000                  MD$DP    000000
A$ALI   000012                   G$$TSS= 000000                   MB$TXO   000000                  MC$P11   000001                  MD$DPV   000044
A$BUF   000006                   G$$TTK= 000000                   MB$ZER   000000                  MC$RFL   002046                  MD$DQ    000006
A$DES   000022                   G$$WRD= 000000                   MC$BAB   006073                  MC$SBU   002051                  MD$DTE   000024
A$FLG   000000                   HEADR = ****** GX                MC$BID   001765                  MC$SFL   002044                  MD$DU    000002
A$LEN   000103                   H.CSP   000000                   MC$BMC   001767                  MC$SYC   000012                  MD$DUP   000012
A$NUM   000010                   H.CUIC  000010                   MC$BSC   001766                  MC$SYL   000036                  MD$DV    000026
A$OPT   000002                   H.DSW   000046                   MC$CAC   001442                  MC$SYR   000050                  MD$DZ    000030
A$PAR   000004                   H.DUIC  000012                   MC$CAP   001440                  MC$SYS   000024                  MD$FUL   000000
A$SCD   000020                   H.EFLM  000004                   MC$CCL   001445                  MC$SYZ   000000                  MD$HAL   000001
A$TYP   000011                   H.EFSV  000040                   MC$CDC   002045                  MC$TBR   001140                  MD$HEL   000002
A$$CHK= 000000                   H.FCS   000050                   MC$CDP   001441                  MC$TBS   001141                  MD$KCP   000013
A$$CPS= 000000                   H.FORT  000052                   MC$CIF   001465                  MC$TDR   002053                  MD$KDP   000034
A$$PRI= 000000                   H.FPSA  000042                   MC$CLD   001464                  MC$TIM   000000                  MD$KDZ   000036
A$$TRP= 000000                   H.FPSL= 000062                   MC$COU   100000                  MC$TYP   007777                  MD$KL    000040
A.ACC   000020                   H.FPVA  000034                   MC$CTL   001454                  MC$UBU   002052                  MD$KMX   000054
A.LNK   000000                   H.GARD  000072                   MC$CTR   001452                  MC$UFD   002047                  MD$KMY   000052
A.NAM   000002                   H.HDLN  000002                   MC$CTS   001453                  MC$UMR   001132                  MD$LEN   000050
A.PUD   000010                   H.IPC   000016                   MC$DOV   002050                  MC$UMS   001133                  MD$PCL   000011
A.REM   000012                   H.IPS   000014                   MC$LBR   001750                  MC$VAX   000003                  MD$QNA   000005
A.UCB   000010                   H.ISP   000020                   MC$LBS   001751                  MC$WID   060000                  MD$UNA   000001
CF$VR2= ****** GX                H.LUN   000076                   MC$LDI   001774                  MC$WIH   040000                  MD$WIT   000002
CHKALI  000036RG                 H.NLUN  000074                   MC$LDO   001775                  MC$WIL   020000                  MD.ABO   000017
CHKPAR  000526RG                 H.NML   000061                   MC$LDR   001762                  MC$W08   020000                  MD.ACC   000010
CHKSCD  000252RG                 H.ODVA  000022                   MC$LDS   001763                  MC$W16   040000                  MD.AOB   000016
CMPCST  000210R                  H.ODVL  000024                   MC$LLB   002021                  MC$W32   060000                  MD.BOB   000011
CURCTX= ****** GX                H.OVLY  000054                   MC$LLP   002115                  MC$XBR   001750                  MD.DIA   000007
C$FLAG= ****** GX                H.PFVA  000032                   MC$LLR   002007                  MC$XBS   001751                  MD.DOB   000015
C$$CKP= 000000                   H.RCVA  000036                   MC$LOO   000001                  MC$XCJ   000322                  MD.DUM   000002
C$$ORE= 000400                   H.RRVA  000062                   MC$LRB   002020                  MC$XCR   002260                  MD.FAI   000014
C$$RSH= 177564                   H.SPRI  000060                   MC$LRP   002114                  MC$XCS   002261                  MD.FNA   000001
DEALUN= ****** GX                H.TKVA  000026                   MC$LRR   002006                  MC$XDR   001762                  MD.FDB   000006
DV.CCL= 000002                   H.TKVL  000030                   MC$LSI   002032                  MC$XDS   001763                  MD.LOA   000001
DV.COM= 020000                   H.VEXT  000056                   MC$LST   002033                  MC$XFR   002272                  MD.LSH   000020
DV.DIR= 000010                   H.WND   000044                   MC$LUP   006072                  MC$XFS   002273                  MD.NOB   000012
DV.EXT= 000400                   H.X25   000064                   MC$MAP   010000                  MC$XLJ   000323                  MD.NON   177777
DV.F11= 040000                   I$$RAR= 000000                   MC$MBL   001764                  MC$XLR   002330                  MD.PER   000000
DV.ISP= 002000                   I$$RDN= 000000                   MC$MBX   004420                  MC$XMA   000310                  MD.RES   000004
DV.MBC= 000400                   K$$CNT= 177546                   MC$MBY   001752                  MC$XMC   002305                  MD.ROB   000005
DV.MNT= 100000                   K$$CSR= 177546                   MC$MPX   004406                  MC$XMS   002304                  MD.RSH   000013
DV.MSD= 000100                   K$$LDC= 000000                   MC$NAP   001604                  MC$XNR   002332                  MD.SDU   000005
DV.OSP= 004000                   K$$TPS= 000074                   MC$NBR   001130                  MC$XRC   002316                  MD.SLO   000003
DV.PSE= 010000                   LC$EXT  000000                   MC$NBS   001131                  MC$XRR   002331                  MD.SNA   000000
DV.REC= 000001                   LC$INT  000001                   MC$NCR   001154                  MC$XRS   002342                  MD.TLD   000004
DV.SDI= 000020                   LD$LP = 000000                   MC$NCS   001155                  MC$020   000002                  MD.UNA   000002
DV.SQD= 000040                   L$$ASG= 000000                   MC$NML   001274                  MD$CI    000007                  MD.UNR   000003
DV.SWL= 001000                   L$$DRV= 000000                   MC$NMR   001142                  MD$CNA   000003                  MD.UOB   000007
DV.TTY= 000004                   L$$P11= 000001                   MC$NMS   001143                  MD$COU   000001                  MD.VOL   000006
DV.UMD= 000200                   L$$11R= 000000                   MC$NNO   001606                  MD$DA    000010                  ME$ALI   000010
D$$BUG= 177514                   MAXACC= 000020                   MC$NNU   001605                  MD$DL    000004                  ME$AL2   000003
D$$ISK= 000000                   MAXID = 000020                   MC$NOP   001607                  MD$DLV   000020                  ME$ARE   000005
D$$L11= 000001                   MAXPAS= 000010                   MC$NOR   000000                  MD$DMC   000014                  ME$CIR   000003
```

```
                                                                      L 4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-2
Symbol table
```

ARREIN - READ AREA STATUS INITIALIZATION

```
    182 000356  000420                  70$:    BR      100$                    ;; SUCCESS !
    183                                 ;
    184                                 ; LEVEL 2 ROUTER: INITIALIZE LIMIT OF SCAN.  CHECK FOR
    185                                 ; NO AREAS IN DATEBASE (UNLIKELY, BUT...)
    186                                 ;
    187 000360  016264  000030  000006  80$:    MOV     N$ROA2(R2),A$MAX(R4)    ;; PICK UP MAXIMUM AREA NUMBER
    188 000366  005364  000006                  DEC     A$MAX(R4)               ;; DON'T COUNT ZERO-TH ENTRY
    189 000372  001012                          BNE     100$                    ;; IF NE, OK
    190 000374  005064  000016                  CLR     A$PAR(R4)               ;; NO DETAIL
    191 000400  012766  177747  000000G         MOV     #ME.OPE,R$R0(SP)        ;; ELSE OPERATION FAILURE
    192
    193 000406                          90$:    RETC    R0                      ;; ELSE SET USER STATE CARRY BIT
    194
    195 000420                          100$:   RETURN                          ;; BACK TO USER STATE
    196 000422  103415                  110$:   BCS     150$                    ; IF CS, ERROR OR
    197
    198 000424  032764  000002  000000  120$:   BIT     #AF.KNO,A$FLG(R4)       ; SINGULAR ENTITY ?
    199 000432  001004                          BNE     130$                    ; IF EQ, KNOWN AREAS
    200 000434  126464  000004  000006          CMPB    A$NUM(R4),A$MAX(R4)     ; AREA NUMBER TOO HIGH FOR SYSTEM ?
    201 000442  003003                          BGT     140$                    ; IF HI, RETURN NO INFO
    202
    203 000444  052764  000001  000000  130$:   BIS     #AF.INI,A$FLG(R4)       ; ELSE REMEMBER SUCCESSFUL INIT
    204
    205 000452  112700  000002          140$:   MOVB    #MS.MOR,R0              ; ELSE MORE TO COME
    206                                 ;
    207                                 ; BUILD RESPONSE MESSAGE
    208                                 ;
    209 000456  012702  000001          150$:   MOV     #1,R2                   ; SINGLE BYTE RETURN
    210 000462  000241                          CLC                             ; ASSUME SUCCESS !
    211 000464  110013                          MOVB    R0,(R3)                 ; STORE THE RETURN CODE
    212 000466  100011                          BPL     170$                    ; IF PL, SUCCESS
    213 000470  116463  000016  000001          MOVB    A$PAR(R4),1(R3)         ; STORE LOW BYTE OF ERROR DETAIL
    214 000476  116463  000017  000002          MOVB    A$PAR+1(R4),2(R3)       ; STORE HIGH BYTE
    215 000504  012702  000003                  MOV     #3,R2                   ; UPDATE MESSAGE LENGTH
    216 000510  000261                  160$:   SEC                             ; ELSE, INDICATE FAILURE
    217 000512                          170$:   RETURN
    218
    219         000001                          .END
```

Symbol table

```
 75                                     .SBTTL  ARREOP - AREA READ STATUS OPERATE
 76                             ;+
 77                             ; *** - ARREOP - AREA READ STATUS OPERATE
 78                             ;
 79                             ; INPUTS:
 80                             ;       R3 = ADDRESS OF OUTPUT BUFFER
 81                             ;       R4 = ADDRESS OF CONTEXT AREA
 82                             ;
 83                             ; OUTPUTS:
 84                             ;       C-BIT = SUCCESS/FAILURE
 85                             ;       R0 = NICE PROTOCOL RETURN CODE
 86                             ;       R2 = LENGTH OF INFORMATION IN OUTPUT BUFFER
 87                             ;       R3-R5 = PRESERVED
 88                             ;
 89                             ;-
 90
 91 000002                      ARREOP::
 92 000002  004567  000000G             JSR     R5,$SAVRG               ; SAVE R3-R5
 93 000006  010364  000002              MOV     R3,A$BUF(R4)            ; SAVE OUTPUT BUFFER ADDRESS
 94 000012  105023                      CLRB    (R3)+                   ; POINT PAST RETURN CODE
 95 000014  032764  000001  000000      BIT     #AF.INI,A$FLG(R4)       ; DONE ?
 96 000022  001562                      BEQ     45$                     ; IF EQ, ALL DONE
 97                             ;
 98                             ; CHECK IF MORE AREAS TO DISPLAY
 99                             ;
100 000024  032764  000002  000000 10$:  BIT     #AF.KNO,A$FLG(R4)       ; KNOWN AREAS ?
101 000032  001406                      BEQ     15$                     ; IF EQ, NO
102
103 000034  005264  000004              INC     A$NUM(R4)               ; ELSE ADVANCE AREA COUNT
104 000040  026464  000004  000006      CMP     A$NUM(R4),A$MAX(R4)     ; PAST MAXIMUM AREAS FOR SYSTEM ?
105 000046  101150                      BHI     45$                     ; IF HI, NO INFORMATION
106
107 000050  005001                  15$:    CLR     R1                      ; NO NEXT NODE
108 000052  005002                          CLR     R2                      ; NO CHANNEL
109 000054                                  SWSTK$  30$                     ;; ENTER SYSTEM STATE
110 000060  016402  000012                  MOV     A$XPT(R4),R2            ;; POINT TO XPT PDV
111 000064  016202  000016                  MOV     Z.DAT(R2),R2            ;; POINT TO PROCESS DDB
112 000070  001456                          BEQ     20$                     ;; IN CASE SYSTEM GOING DOWN
113                             ;
114                             ; EXECUTOR AREA
115                             ;
116 000072  026464  000004  000014          CMP     A$NUM(R4),A$EXA(R4)     ;; IS THIS THE EXECUTOR AREA ?
117 000100  001010                          BNE     17$                     ;; IF NE, NO
118 000102                                  CALL    GETEXA                  ;; GET EXECUTOR CHANNEL AND NODE
119 000106  103447                          BCS     20$                     ;; IF CS, ERROR
120 000110  010166  000000G                 MOV     R1,R$R1(SP)             ;; ELSE PASS BACK NODE ADDRESS
121 000114  010266  000000G                 MOV     R2,R$R2(SP)             ;; CHANNEL NUMBER OF LINE
122 000120  000447                          BR      25$                     ;; SUCCESS
123                             ;
124                             ; REMOTE AREA
125                             ;
126 000122                          17$:    MAP     N$ROA2+2(R2)            ;; MAP TO LEVEL 2 REACHABILITY TABLES
127 000130  001436                          BEQ     20$                     ;; IF EQ, NOT THERE
128 000132  016200  000034                  MOV     N$ROA2+4(R2),R0         ;; POINT TO THEM
129 000136                                  BIAS    R0                      ;; BIAS ADDRESS FOR APR 6
130 000146  016401  000004                  MOV     A$NUM(R4),R1            ;; GET AREA NUMBER
131 000152  006301                          ASL     R1                      ;; MAKE AREA NUMBER A WORD OFFSET
```

```
                                                                  K  7
ARREOP - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 11-4
Symbol table

SF.LPB= 000004          S.LEN   000004          ZF.DVP= 100000          ZF.X3P= 000000          Z.PCB   000012
SF.MFL= 000040          S.NMST  000002          ZF.INI= 040000          ZS.ASN= 100000          Z.SCH   000007
SF.PAC= 000020          S.OWNR  000003          ZF.KMX= 000020          ZS.BSY= 140000          $CBTA = ****** GX
SF.REA= 000010          TEMP    000000R         ZF.LLC= 000004          Z.AVL   000014          $C5TA = ****** GX
SF.SER= 000001          T$$KMG= 000000          ZF.LMC= 000100          Z.DAT   000016          $HEADR= ****** GX
SF.SVC= 000002          T$$MIN= 000000          ZF.MAN= 020000          Z.DSP   000000          $SAVRG= ****** GX
SF.UNL= 000040          V$$CTR= 001000          ZF.MFL= 000010          Z.FLG   000010          $SAVVR= ****** GX
SLTMA = ****** GX       X$$DBT= 000000          ZF.MTM= 000400          Z.LEN = 000016          $$$SYS= 004374
S$$WRG= 000000          ZF.COU= 001000          ZF.MUX= 000040          Z.LLN   000006          $$$SYX= 000000
S$$YSZ= 007600          ZF.DDM= 000001          ZF.PSE= 002000          Z.MAP   000020          .MBILD= ****** GX
S.COST  000001          ZF.DIA= 004000          ZF.SLI= 010000          Z.NAM   000004          .MCODE= ****** GX
S.FLG   000000          ZF.DLC= 000002          ZF.TIM= 000200

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        001530    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 231
Work  file writes: 162
Size of work file: 31975 Words  ( 125 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:36.07
SY:ARREOP.V2,[135,134]ARREOP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ARREOP
```

```
                                                                 L  7
ARREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:08      PAGE 1

SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
                                            .SBTTL  BIT DEFINITIONS
                                    ;
                                    ; PARSE FLAG DEFINITIONS (L$PFG)
                                    ;
         000001                             LP$UNT = 1                      ; UNIT NUMBER FOUND
         000002                             LP$TRB = 2                      ; TRIBUTARY NUMBER FOUND
         000004                             LP$MUX = 4                      ; DEVICE IS MUX
         000010                             LP$MPT = 10                     ; LINE IS MULTIPOINT
         000020                             LP$WDV = 20                     ; WILD CARD DEVICE NAME FOUND
         000040                             LP$WCN = 40                     ; WILD CARD CONTROLLER NUMBER FOUND
         000100                             LP$WUN = 100                    ; WILD CARD UNIT NUMBER FOUND
         000200                             LP$WTR = 200                    ; WILD CARD TRIBUTARY NUMBER FOUND
         000360                             LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK
                                    ;
                                    ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                    ;
         000001                             LF$REA = 1                      ; READ COUNTERS OPERATION
         000002                             LF$ZER = 2                      ; ZERO COUNTERS OPERATION
         000004                             LF$SKP = 4                      ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                            ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                            ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                            ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                            ; OF THE TRIBUTARY COUNTERS.
         000010                             LF$VR2 = 10                     ; CONNECTED TO VERSION 2.0 NCP
         000020                             LF$END = 20                     ; EXECUTOR IS AN ENDNODE
         000040                             LF$SIG = 40                     ; SHOWING SIGNIFIGANT LINES/CIRCUITS
         040000                             LF$MLT = 40000                  ; MULTIPLE ADJACENCY FLAG
         100000                             LF$SEG  = 100000                ; SEGMENTED RESPONSE IN PROGRESS
                                    ;
                                    ; BIT POSITION DEFINITIONS
                                    ;
         000001                             BIT0 = 1
         000002                             BIT1 = 2
         000004                             BIT2 = 4
         000010                             BIT3 = 10
         000020                             BIT4 = 20
         000040                             BIT5 = 40
         000100                             BIT6 = 100
         000200                             BIT7 = 200
         000400                             BIT8 = 400
         001000                             BIT9 = 1000
         002000                             BIT10 = 2000
         004000                             BIT11 = 4000
         010000                             BIT12 = 10000
         020000                             BIT13 = 20000
         040000                             BIT14 = 40000
         100000                             BIT15 = 100000

                                    ;
                                    ; DEFINE LOCAL FLAGS AND OFFSETS
                                    ;
         000004                             NF.WLD  = NF.LPB        ; REUSE FLAG BIT
```

```
                                                                 K  9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-3
Symbol table

AT$ACL= 000100            DISTBL  000000R           E$NNOD  000002            LF$MLT= 040000            MAPXPT= ****** GX
AT$ADN= 000040            D$AMXC  000072            E$NRT   000042            LF$REA= 000001            MAXLEN= 000006
AT$AUP= 000020            D$AMXH  000074            E$NRTP  000005            LF$SEG= 100000            MB$DIS  000001
AT$CYC= 000004            D$ANN   000000            E$NSEG  000010            LF$SIG= 000040            MB$ENA  000000
AT$LV1= 000002            D$BRPR  000102            E$NTIM  000046            LF$SKP= 000004            MB$FUL  000002
AT$LV2= 000001            D$BRTM  000100            E$NUSE  000004            LF$VR2= 000010            MB$MIX  000002
AT$NEX= 000200            D$DELF  000045            E$STRT  000006            LF$ZER= 000002            MB$ONE  000001
AT$UP = 000010            D$DELW  000046            E$$XPR= 000000            LP$MPT= 000010            MB$RXO  000001
A$CIR   000003            D$END = 000104            FW.MAX= 000040            LP$MUX= 000004            MB$TXO  000000
A$LEN   000004            D$FNB   000034            FW.RSV= 000300            LP$TRB= 000002            MB$ZER  000000
A$NID   000000            D$HIOR  000024            F$$LVL= 000001            LP$UNT= 000001            MC$BAB  006073
A$TM    000003            D$HOST  000022            G$$TPP= 000000            LP$WCN= 000040            MC$BID  001765
A$TMI   000002            D$INAC  000044            G$$TSS= 000000            LP$WDV= 000020            MC$BMC  001767
A$TSZ   000000            D$INCT  000042            G$$TTK= 000000            LP$WLD= 000360            MC$BSC  001766
A$TYP   000002            D$IPL   000051            G$$WRD= 000000            LP$WTR= 000200            MC$CAC  001442
A$$CHK= 000000            D$LID   000020            H.CSP   000000            LP$WUN= 000100            MC$CAP  001440
A$$CPS= 000000            D$LNAM  000006            H.CUIC  000010            L$ADJ   000110            MC$CCL  001445
A$$PRI= 000000            D$LNUM  000014            H.DSW   000046            L$BUF   000060            MC$CDC  002045
A$$TRP= 000000            D$LST   000047            H.DUIC  000012            L$CHN   000054            MC$CDP  001441
BIT0  = 000001            D$MAXC  000064            H.EFLM  000004            L$COU   000114            MC$CIF  001465
BIT1  = 000002            D$MAXH  000066            H.EFSV  000040            L$CTB   000053            MC$CLD  001464
BIT10 = 002000            D$MAXV  000070            H.FCS   000050            L$CTL   000002            MC$COU  100000
BIT11 = 004000            D$MLL   000040            H.FORT  000052            L$DDM   000000            MC$CTL  001454
BIT12 = 010000            D$MNOD  000041            H.FPSA  000042            L$FLG   000064            MC$CTR  001452
BIT13 = 020000            D$NA    000062            H.FPSL= 000062            L$FLX   000070            MC$CTS  001453
BIT14 = 040000            D$NBEA  000056            H.FPVA  000034            L$LEN   000124            MC$DOV  002050
BIT15 = 100000            D$NBRA  000054            H.GARD  000072            L$LTM   000070            MC$LBR  001750
BIT2  = 000004            D$NEND= 000054            H.HDLN  000002            L$MSG   000056            MC$LBS  001751
BIT3  = 000010            D$NLN   000030            H.IPC   000016            L$MTYP  000067            MC$LDI  001774
BIT4  = 000020            D$NN    000060            H.IPS   000014            L$NAM   000000            MC$LDO  001775
BIT5  = 000040            D$OUTT  000043            H.ISP   000020            L$NLEN= 000020            MC$LDR  001762
BIT6  = 000100            D$RETF  000050            H.LUN   000076            L$NOD   000112            MC$LDS  001763
BIT7  = 000200            D$RNN   000002            H.NLUN  000074            L$NXT   000044            MC$LLB  002021
BIT8  = 000400            D$RTMR  000076            H.NML   000061            L$OPT   000062            MC$LLP  002115
BIT9  = 001000            D$SEG   000036            H.ODVA  000022            L$PAR   000120            MC$LLR  002007
BLNLST  000054RG          D$SER   000032            H.ODVL  000024            L$PDV   000055            MC$LOO  000001
B$10    000003            D$SQRL  000052            H.OVLY  000054            L$PFG   000041            MC$LRB  002020
B$11    000004            D$$BUG= 177514            H.PFVA  000032            L$PLB   000116            MC$LRP  002114
B$12    000005            D$$IJK= 000000            H.RCVA  000036            L$PRO   000067            MC$LRR  002006
B$5     000000            D$$L11= 000001            H.RRVA  000062            L$PVC   000004            MC$LSI  002032
B$6     000001            D$$YNC= 000000            H.SPRI  000060            L$SCN   000020            MC$LST  002033
B$7     000002            D$$YNM= 000000            H.TKVA  000026            L$SCR   000072            MC$LUP  006072
B$7S10  000006            D.DID   000003            H.TKVL  000030            L$SLEN  000066            MC$MAP  010000
CEACC = ****** GX         D.FLG   000000            H.VEXT  000056            L$SLT   000042            MC$MBL  001764
CF.INI= 000001            D.SID   000013            H.WND   000044            L$SNM   000046            MC$MBX  004420
CF.NET= 000015            D.VIS   000022            H.X25   000064            L$SYL   000122            MC$MBY  001752
CF.NRT= 000013            E$NBR   000014            I$$RAR= 000000            L$TNM   000052            MC$MPX  004406
CF.RSV= 000360            E$NBS   000020            I$$RDN= 000000            L$TPT   000050            MC$NAP  001604
CF.RT1= 000007            E$NCR   000034            KISAR6= ****** GX         L$TRB   000040            MC$NBR  001130
CF.RT2= 000011            E$NCS   000036            K$$CNT= 177546            L$TSZ   000070            MC$NBS  001131
CF.TST= 000005            E$NIC   000044            K$$CSR= 177546            L$TYP   000063            MC$NCR  001154
CF.TYP= 000016            E$NLEN  000050            K$$LDC= 000000            L$UNT   000003            MC$NCS  001155
CF.VFY= 000003            E$NLLA  000012            K$$TPS= 000074            L$$ASG= 000000            MC$NML  001274
C$$CKP= 000000            E$NLNK  000000            LC$EXT  000000            L$$DRV= 000000            MC$NMR  001142
C$$ORE= 000400            E$NML   000040            LC$INT  000001            L$$P11= 000001            MC$NMS  001143
C$$RSH= 177564            E$NMR   000024            LD$LP = 000000            L$$11R= 000000            MC$NNO  001606
DECPT = ****** GX         E$NMS   000030            LF$END= 000020            MAPECL= ****** GX         MC$NNU  001605
```

```
                                                                 L  9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-4
Symbol table
```

Table of contents

```
     5-   55    MACRO CALLS AND LOCAL DATA
     6-   63    BIT DEFINITIONS
     7-   66    CHKCIR - CHECK CIRCUIT-ID FORMAT
```

```
**FILE**ID**CHKLIN
```

```
  CCCCCCCC   HH      HH   KK      KK    LL             IIIIII     NN      NN
  CCCCCCCC   HH      HH   KK      KK    LL             IIIIII     NN      NN
CC           HH      HH   KK      KK    LL               II       NN      NN
CC           HH      HH   KK      KK    LL               II       NN      NN
CC           HH      HH   KK    KK      LL               II       NNNN    NN
CC           HH      HH   KK    KK      LL               II       NNNN    NN
CC           HHHHHHHHHH   KKKKKK        LL               II       NN  NN  NN
CC           HHHHHHHHHH   KKKKKK        LL               II       NN  NN  NN
CC           HH      HH   KK    KK      LL               II       NN    NNNN
CC           HH      HH   KK    KK      LL               II       NN    NNNN
CC           HH      HH   KK      KK    LL               II       NN      NN       ....
CC           HH      HH   KK      KK    LL               II       NN      NN       ....
  CCCCCCCC   HH      HH   KK      KK    LLLLLLLLLL     IIIIII     NN      NN       ....
  CCCCCCCC   HH      HH   KK      KK    LLLLLLLLLL     IIIIII     NN      NN       ....
```

```
LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
CHKLIN - CHECK A LINE-ID FORMAT MACRO V05.03b Saturday 29-Jun-85 17:42
Table of contents
```

K 12

```
CHKLIN     CREATED BY  MACRO  ON 29-JUN-85 AT 17:42     PAGE 3
MACRO CROSS REFERENCE                                   CREF   04.00
MACRO NAME        REFERENCES

CALL                7-103
CIRCX$            #5-59         5-63
MANDF$            #5-59         5-61
RETURN              7-109
```

L 12

```
**FILE**ID**CICHIN
```

```
    210                                                 .SBTTL  PARSE SUB-EXPRESSIONS
    211                                         .IFT
    212                                         ;
    213                                         ; SET ALL SUB-EXPRESSION - INVALID OPTION
    214                                         ;
    215 000000                                          FIELD$  SALL
    216 000000                                           $EOM   ,SALL1
    217
    218 000000                                          FIELD$  SALL1
    219 000000                                           $ERROR  ME.FUN
    220
    221                                         .IFTF
    222                                         ;
    223                                         ; STATE SUB-EXPRESSION
    224                                         ;
    225 000000                                          FIELD$  SSTA
    226 000000                                           $PARAM MP$STA,,PAREND,MP$STA,+L$PAR
    227 000000                                          FIELD$
    228 000000                                           $BYTE   MS$ON,$EXIT,,MS$ON,+L$STA
    229 000000                                           $BYTE   MS$OFF,$EXIT,,MS$OFF,+L$STA
    230                                         .IFT
    231 000000                                           $BYTE   MS$SER,$EXIT,,MS$SER,+L$STA
    232                                         .IFTF
    233 000000                                           $NULL   ,,PVAERR
    234                                         .IFT
    235                                         ;
    236                                         ; COST SUB-EXPRESSION
    237                                         ;
    238 000000                                          FIELD$  SCOS
    239 000000                                           $PARAM MP$COS,,PAREND,MP$COS,+L$PAR
    240 000000                                          FIELD$
    241 000000                                           $SKIP   ,$EXIT,COSACT
    242                                         ;
    243                                         ; Service subexpression
    244                                         ;
    245 000000                                          FIELD$  SSER
    246 000000                                           $PARAM MP$SER,,PAREND,MP$SER,+L$PAR
    247 000000                                          FIELD$
    248 000000                                           $BYTE   MB$ENA,$EXIT,,MB$ENA,+L$SER
    249 000000                                           $BYTE   MB$DIS,$EXIT,,MB$DIS,+L$SER
    250                                         ;
    251                                         ; COUNTER TIMER SUB-EXPRESSION
    252                                         ;
    253 000000                                          FIELD$  SLCT
    254 000000                                           $PARAM MP$LCT,,PAREND,MP$LCT,+L$PAR
    255 000000                                          FIELD$
    256 000000                                           $MATCH 2,$EXIT,LCTACT
    257 000000                                           $NULL   ,,PVAERR
    258                                         ;
    259                                         ; HELLO TIMER SUB-EXPRESSION
    260                                         ;
    261 000000                                          FIELD$  SHTM
    262 000000                                           $PARAM  MP$HTM,,PAREND,MP$HTM,+L$PAR
    263 000000                                          FIELD$
    264 000000                                           $MATCH  2,$EXIT,HTMACT
    265 000000                                           $NULL    ,,PVAERR
    266                                         ;
```

```
                                          .SBTTL  PNAERR - PARAMETER NOT APPLICABLE ERROR RETURN
                                    .IFT
                                          .SBTTL  PVAERR - PARAMETER VALUE ERROR RETURN
                                          .SBTTL  CIRERR - CIRCUIT-ID FORMAT ERROR
                                    .IFTF
                                  ;+
                                  ; **-PNAERR-PARAMETER NOT APPLICABLE ERROR RETURN
                                    .IFT
                                  ; **-PVAERR-PARAMETER VALUE ERROR RETURN
                                  ; **-CIRERR-CIRCUIT-ID FORMAT ERROR RETURN
                                    .IFTF
                                  ;
                                  ; THESE ROUTINES TERMINATE THE PARSE WITH THE APPROPRIATE
                                  ; ERROR CODE RETURNED IN R0.
                                  ;
                                  ;-
                                          .ENABL  LSB
                                    .IFT
000462  012700  177752            PNAERR: MOV     #ME.PNA,R0              ; SET PARAM NOT APPLICABLE ERROR
000466  000416                            BR      10$                     ; AND TERMINATE PARSE
                                    .IFTF
000470  012700  177760            PVAERR: MOV     #ME.PVA,R0              ; SET PARAM VALUE ERROR
                                    .IFT
000474  000413                            BR      10$                     ; AND TERMINATE PARSE

000476  012700  177767            CIRERR: MOV     #ME.IID,R0              ; SET INVALID CIRCUIT-ID ERROR
000502  012764  000003  000072            MOV     #ME$CIR,L$PAR(R4)       ;  ...
000510  000405                            BR      10$                     ; JOIN COMMON CODE
                                    .IFTF

000512  012700  177765            CSTERR: MOV     #ME.CST,R0              ; COMPONENT IN WRONG STATE
000516  012764  000003  000072            MOV     #ME$CIR,L$PAR(R4)       ; CIRCUIT

000524  016706  000000G           10$:    MOV     .MSGST,SP               ; RESTORE SAVED SP
000530  012601                            MOV     (SP)+,R1                ; RESTORE R1
000532  000261                            SEC                             ; SET ERROR RETURN
000534                                    RETURN

                                          .DSABL  LSB
```

```
                                        .SBTTL  PLTACT - POLL TIMER ACTION ROUTINE

                                ;+
                                ;
                                ; PLTACT - POLL TIMER ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 2 BYTE POLL TIMER
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$PLT(R4) - POLL TIMER VALUE
                                ;
                                ;-
001176                          PLTACT: CALL    CHKDMP                  ; CHECK FOR DMP
001202  010164  000112                  MOV     R1,L$PLT(R4)            ; SAVE POLL TIMER VALUE
001206                                  RETURN
```

```
                                   .SBTTL  MXBACT - MAXIMUM BLOCKS ACTION ROUTINE

                           ;+
                           ;
                           ; MXBACT - MAXIMUM BLOCKS ACTION ROUTINE
                           ;
                           ; INPUTS:
                           ;         .MSGPT - POINTER TO 1 BYTE MAXIMUM BLOCKS
                           ;         R4     - POINTER TO CONTEXT AREA
                           ;
                           ; OUTPUTS:
                           ;         L$MXB(R4) - MAXIMUM BLOCKS VALUE
                           ;
                           ;-
001400                     MXBACT: CALL     CHKDMP                  ; CHECK FOR DMP
001404  110164  000133             MOVB     R1,L$MXB(R4)            ; SAVE MAXIMUM BLOCKS VALUE
001410                             RETURN
```

```
     000362  026164  000010  000020            CMP     A.UCB(R1),A$SCO(R4)      ;; ELSE, IS THIS ONE TO BE DISPLAYED ?

     000370  001363                            BNE     30$                      ;; IF NE, NO
     000372                            40$:    CALL    SAVCTX                   ;; ELSE, SAVE ALIAS CONTEXT
     000376  103360                            BCC     30$                      ;; IF CC, BR

     000400                            80$:    RESMAP                           ;; RESTORE MAPPING
     000404                            90$:    RETURN                           ;; BACK TO USER STATE AND RETURN
```

```
                                                          L 2
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 5-1
MACRO CALLS AND LOCAL DATA

     78                                         .EVEN
     79                                         .LIST   BEX
```

```
                                                          M 2
ALREOP - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:06  Page 6
ALREOP - ALIAS READ INFORMATION OPERATE
```

```
                                                                L 3
ALREOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:06      PAGE 2
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME        REFERENCES

ALRCX$            #5-57       5-64
ANBDF$            #5-56       5-59
CALL               6-142      6-147      6-148      8-239      9-277      9-281      9-289      9-290      9-300      10-341
DHBDF$            #5-57       5-62
HDRDF$            #5-57       5-61
MANDF$            #5-56       5-60
MAP               #5-56
RESMAP            #5-56       9-302
RETC              #5-56       9-283
RETURN             6-163      7-198      8-245      9-303      9-315      10-355
SAVMAP            #5-56       9-278
SWSTK$            #5-56       9-277
```

```
                                                                M 3
**FILE**ID**ALXXIN
```

```
                                                                                L 4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-2
Symbol table

ME$EXA  000000          MF$TRA  000301          MP$AMC  001640          MP$DLT  002200          MP$MAV  006321
ME$EXE  000200          MF$TRI  000021          MP$AMH  001641          MP$DRO  001441          MP$MBN  001636
ME$LIN  000001          MF$ZER  000025          MP$ANB  004420          MP$DST  000454          MP$MBR  001637
ME$LOG  000002          ML$ALL  100000          MP$ASB  000024          MP$DTE  002140          MP$MBU  001642
ME$MOD  000004          ML$CLS  000000          MP$ASC  004432          MP$DTY= 001452          MP$MCB  000156
ME$NOD  000000          ML$CON  000001          MP$AUS  000512          MP$DUA  000207          MP$MCO  001632
ME$NON  177777          ML$EXT  000001          MP$BBT  002165          MP$DUC  000210          MP$MDE  004421
ME$OBJ  000007          ML$FIL  000002          MP$BDF  002164          MP$DUM  000202          MP$MHO  001633
ME$OB2  000004          ML$FIR  000000          MP$BFQ  002121          MP$DUP  002127          MP$MLB  000202
ME$PRC  000005          ML$INT  000000          MP$BLK  001616          MP$DVC  002114          MP$MLK  001306
ME$SYS  000006          ML$KNO  140000          MP$BLO  001452          MP$DWE  001321          MP$MLN  001631
ME.BLO  177744          ML$MON  000003          MP$BMX  002176          MP$ELT  000157          MP$MLP  006333
ME.CON  177753          ML$SYS  000002          MP$BNP  006330          MP$ETY  001605          MP$MRB  002171
ME.CST  177765          ML$TOP  000001          MP$BRT  001620          MP$EVE  000311          MP$MRO  001605
ME.DIS  177755          MN$UNL  000377          MP$BSA  002176          MP$FNC  001752          MP$MRP  006322
ME.DON  177600          MO$ACC  000200          MP$BSD  002203          MP$GDT  002222          MP$MRT  002153
ME.FCO  177762          MO$ADD  000002          MP$BSI  002200          MP$GNM  002223          MP$MRV  006320
ME.FIO  177756          MO$ALA  000004          MP$BSP  006331          MP$GRO  002115          MP$MSB  000170
ME.FOP  177763          MO$ALI  000000          MP$BUF  000170          MP$GRP  000541          MP$MVE  000145
ME.FOR  177776          MO$CHA  000040          MP$BUP  006324          MP$GTY  002224          MP$MVI  001634
ME.FUN  177777          MO$CIR  000003          MP$BUS  001643          MP$HAD  001757          MP$MVR  001751
ME.GRO  177745          MO$CLE  000100          MP$CAC  001750          MP$HBT  002142          MP$MWN  002154
ME.HAR  177750          MO$COU  000060          MP$CAS  001762          MP$HDD  000162          MP$MXB  002172
ME.IID  177767          MO$DAC  000000          MP$CAT  002210          MP$HTM  001612          MP$MXC  000466
ME.LCO  177766          MO$DEF  000000          MP$CCS  004406          MP$HWA  002210          MP$MXR  001630
ME.LPR  177757          MO$DPR  000000          MP$CHN  002141          MP$IAT  001322          MP$MXW  002143
ME.MPR  177773          MO$ENT  000017          MP$CIR  000144          MP$IDE  000144          MP$NAA  006335
ME.MVE  177771          MO$EVE  000100          MP$CLK  002131          MP$IDP  006327          MP$NAC  001130
ME.OPE  177747          MO$INF  000160          MP$CLN  002126          MP$IHO  000215          MP$NAP  006326
ME.PLO  177751          MO$INS  000002          MP$CLT  002211          MP$INA  002177          MP$NCT  000240
ME.PMI  177743          MO$LIN  000001          MP$CMB  002142          MP$IND  002204          MP$NET  002114
ME.PNA  177752          MO$LOG  000001          MP$CMK  000537          MP$INI  002201          MP$NLI  000765
ME.PRI  177775          MO$MIR  000031          MP$CMX  002153          MP$ITI  000776          MP$NNA  000764
ME.PTY  177772          MO$NAM  000001          MP$CND  000310          MP$LAA  000231          MP$NOD  000500
ME.PVA  177760          MO$NIC  000023          MP$CNU  001753          MP$LAN  000234          MP$NRA  006334
ME.RES  177761          MO$NOD  000000          MP$COB  000311          MP$LAR  000202          MP$NRB  000214
ME.ROO  177754          MO$OFF  000001          MP$CON  002126          MP$LAS  000012          MP$NSA  001617
ME.SIZ  177774          MO$ON   000000          MP$COS  001604          MP$LBS  000203          MP$NTI  002141
ME.SYS  177746          MO$OPT  000001          MP$CPF  000230          MP$LCO  000226          MP$NUM  001642
ME.UCO  177770          MO$PER  000200          MP$CPL  000232          MP$LCT  000156          MP$NVE  001274
MF$ACT  177776          MO$PRO  000002          MP$CPT  000226          MP$LHL  000232          MP$NXN= 001476
MF$ADD  000000          MO$PRS  000001          MP$CPU  000161          MP$LLE  000227          MP$OAC  000632
MF$ADJ  177774          MO$REA  000200          MP$CSZ  001755          MP$LLO  004432          MP$OAN  000620
MF$ALL  177775          MO$SET  000000          MP$CUS  002127          MP$LMB  002152          MP$OCO  000776
MF$BYE  000300          MO$STA  000020          MP$CVA  000540          MP$LMX  002200          MP$OHO  000214
MF$CHA  000023          MO$SUM  000000          MP$DAL  002570          MP$LNA  000144          MP$ONA  000764
MF$DUM  000020          MO$TSK  000000          MP$DCO= 001464          MP$LNO  000233          MP$ONR  002114
MF$EVT  000001          MO$VOL  000000          MP$DDT  002177          MP$LOA  000170          MP$OQL  001453
MF$FUP  000264          MO$ZER  000000          MP$DEL  001131          MP$LOG  004514          MP$OTI  000777
MF$KNO  177777          MP$ACB  000012          MP$DES  000156          MP$LOO  000620          MP$OTY  001022
MF$LOA  000017          MP$ACC  000514          MP$DEV  002114          MP$LTM  001613          MP$OUS  000777
MF$LOO  177775          MP$ACT  004526          MP$DFA  001320          MP$LTY  004533          MP$OVE  001010
MF$REA  000024          MP$ADD  000766          MP$DHO= 001465          MP$LWI  000230          MP$OWN  004374
MF$SIG  177773          MP$ADJ  001440          MP$DIA  000173          MP$MAC  004420          MP$PAR  000036
MF$SPF  000302          MP$ADP  006325          MP$DLB  002571          MP$MAD  001630          MP$PAS  000513
MF$SYS  000026          MP$ADS  004406          MP$DLG  004521          MP$MAP  006323          MP$PCH  002152
MF$TES  000022          MP$ALB  000036          MP$DLI= 001466          MP$MAR  001635          MP$PCO  000024
```

```
                                                                                M 4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-3
Symbol table
```

```
                                                          L 5
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 6-3
Symbol table

AF.END= 000040            D$LNUM  000014            H.SPRI  000060            MC$LST  002033            MC$XFR  002272
AF.INI= 000001            D$LST   000047            H.TKVA  000026            MC$LUP  006072            MC$XFS  002273
AF.KNO= 000002            D$MAXC  000064            H.TKVL  000030            MC$MAP  010000            MC$XLJ  000323
AF.LV2= 000100            D$MAXH  000066            H.VEXT  000056            MC$MBL  001764            MC$XLR  002330
AF.MUX= 000020            D$MAXV  000070            H.WND   000044            MC$MBX  004420            MC$XMA  000310
ARREIN  000000RG          D$MLL   000040            H.X25   000064            MC$MBY  001752            MC$XMC  002305
A$BUF   000002            D$MNOD  000041            I$$RAR= 000000            MC$MPX  004406            MC$XMS  002304
A$EXA   000014            D$NA    000062            I$$RDN= 000000            MC$NAP  001604            MC$XNR  002332
A$FLG   000000            D$NBEA  000056            K$$CNT= 177546            MC$NBR  001130            MC$XRC  002316
A$LENX  000027            D$NBRA  000054            K$$CSR= 177546            MC$NBS  001131            MC$XRR  002331
A$MAX   000006            D$NEND= 000054            K$$LDC= 000000            MC$NCR  001154            MC$XRS  002342
A$NAM   000020            D$NLN   000030            K$$TPS= 000074            MC$NCS  001155            MC$020  000002
A$NUM   000004            D$NN    000060            LC$EXT  000000            MC$NML  001274            MD$CI   000007
A$OPT   000010            D$OUTT  000043            LC$INT  000001            MC$NMR  001142            MD$CNA  000003
A$PAR   000016            D$RETF  000050            LD$LP = 000000            MC$NMS  001143            MD$COU  000001
A$XPT   000012            D$RNN   000002            L$$ASG= 000000            MC$NNO  001606            MD$DA   000010
A$$CHK= 000000            D$RTMR  000076            L$$DRV= 000000            MC$NNU  001605            MD$DL   000004
A$$CPS= 000000            D$SEG   000036            L$$P11= 000001            MC$NOP  001607            MD$DLV  000020
A$$PRI= 000000            D$SER   000032            L$$11R= 000000            MC$NOR  000000            MD$DMC  000014
A$$TRP= 000000            D$SQRL  000052            MB$DIS  000001            MC$NPF  001616            MD$DMF  000046
C$$CKP= 000000            D$$BUG= 177514            MB$ENA  000000            MC$NPR  001630            MD$DMP  000022
C$$ORE= 000400            D$$ISK= 000000            MB$FUL  000002            MC$NRE  001200            MD$DMR  000050
C$$RSH= 177564            D$$L11= 000001            MB$MIX  000002            MC$NRT  001166            MD$DMV  000042
DECPT = ****** GX         D$$YNC= 000000            MB$ONE  000001            MC$NVR  001642            MD$DN   000016
DV.CCL= 000002            D$$YNM= 000000            MB$RXD  000001            MC$P08  000000            MD$DP   000000
DV.COM= 020000            E$$XPR= 000000            MB$TXD  000000            MC$P11  000001            MD$DPV  000044
DV.DIR= 000010            F$$LVL= 000001            MB$ZER  000000            MC$RFL  002046            MD$DQ   000006
DV.EXT= 000400            G$$TPP= 000000            MC$BAB  006073            MC$SBU  002051            MD$DTE  000024
DV.F11= 040000            G$$TSS= 000000            MC$BID  001765            MC$SFL  002044            MD$DU   000002
DV.ISP= 002000            G$$TTK= 000000            MC$BMC  001767            MC$SYC  000012            MD$DUP  000012
DV.MBC= 000400            G$$WRD= 000000            MC$BSC  001766            MC$SYL  000036            MD$DV   000026
DV.MNT= 100000            HEADR = ****** GX         MC$CAC  001442            MC$SYR  000050            MD$DZ   000030
DV.MSD= 000100            H.CSP   000000            MC$CAP  001440            MC$SYS  000024            MD$FUL  000000
DV.OSP= 004000            H.CUIC  000010            MC$CCL  001445            MC$SYZ  000000            MD$HAL  000001
DV.PSE= 010000            H.DSW   000046            MC$CDC  002045            MC$TBR  001140            MD$HEL  000002
DV.REC= 000001            H.DUIC  000012            MC$CDP  001441            MC$TBS  001141            MD$KCP  000013
DV.SDI= 000020            H.EFLM  000004            MC$CIF  001465            MC$TDR  002053            MD$KDP  000034
DV.SQD= 000040            H.EFSV  000040            MC$CLD  001464            MC$TIM  000000            MD$KDZ  000036
DV.SWL= 001000            H.FCS   000050            MC$COU  100000            MC$TYP  007777            MD$KL   000040
DV.TTY= 000004            H.FORT  000052            MC$CTL  001454            MC$UBU  002052            MD$KMX  000054
DV.UMD= 000200            H.FPSA  000042            MC$CTR  001452            MC$UFD  002047            MD$KMY  000052
D$AMXC  000072            H.FPSL= 000062            MC$CTS  001453            MC$UMR  001132            MD$LEN  000050
D$AMXH  000074            H.FPVA  000034            MC$DOV  002050            MC$UMS  001133            MD$PCL  000011
D$ANN   000000            H.GARD  000072            MC$LBR  001750            MC$VAX  000003            MD$QNA  000005
D$BRPR  000102            H.HDLN  000002            MC$LBS  001751            MC$WID  060000            MD$UNA  000001
D$BRTM  000100            H.IPC   000016            MC$LDI  001774            MC$WIH  040000            MD$WIT  000002
D$DELF  000045            H.IPS   000014            MC$LDD  001775            MC$WIL  020000            MD.ABO  000017
D$DELW  000046            H.ISP   000020            MC$LDR  001762            MC$W08  020000            MD.ACC  000010
D$END = 000104            H.LUN   000076            MC$LDS  001763            MC$W16  040000            MD.AOB  000016
D$FNB   000034            H.NLUN  000074            MC$LLB  002021            MC$W32  060000            MD.BOB  000011
D$HIOR  000024            H.NML   000061            MC$LLP  002115            MC$XBR  001750            MD.DIA  000007
D$HOST  000022            H.ODVA  000022            MC$LLR  002007            MC$XBS  001751            MD.DOB  000015
D$INAC  000044            H.ODVL  000024            MC$LOD  000001            MC$XCJ  000322            MD.DUM  000002
D$INCT  000042            H.OVLY  000054            MC$LRB  002020            MC$XCR  002260            MD.FAI  000014
D$IPL   000051            H.PFVA  000032            MC$LRP  002114            MC$XCS  002261            MD.FNA  000001
D$LID   000020            H.RCVA  000036            MC$LRR  002006            MC$XDR  001762            MD.FOB  000006
D$LNAM  000006            H.RRVA  000062            MC$LSI  002032            MC$XDS  001763            MD.LOA  000001
```

```
                                                          M 5
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 6-4
Symbol table
```

ARREOP - AREA READ STATUS OPERATE

```
000154  060100                          ADD     R1,R0                   ;; POINT TO REACHABILITY INFORMATION
000156  011000                          MOV     (R0),R0                 ;; PICK UP REACHABILITY INFO
000160  001422                          BEQ     20$                     ;; IF EQ, NOT REACHABLE
000162                                  MAP     N$ADJ1(R2)              ;; MAP TO THE ADJACENCY DATABASE
000170                                  BIAS    R0                      ;; BIAS ADJACENCY ADDRESS
000200  132760  000010  000002          BITB    #AT$UP,A$TYP(R0)        ;; IS THIS ADJACENCY UP ?
000206  001407                          BEQ     20$                     ;; IF EQ, NO
000210  016066  000000  000000G         MOV     A$NID(R0),R$R1(SP)      ;; PICK UP NEXT NODE IN PATH
000216  116066  000003  000000G         MOVB    A$CIR(R0),R$R2(SP)      ;; ... CHANNEL NUMBER
000224  000405                          BR      25$                     ;; DONE

000226                          20$:    RETC    R0                      ;; SET USER CARRY - AREA NOT KNOWN
000240                          25$:    RETURN                          ;; BACK TO USER STATE

000242  103007                  30$:    BCC     32$                     ; IF CC, GOT A LIVE ONE
000244  032764  000002  000000          BIT     #AF.KNO,A$FLG(R4)       ; ELSE, ARE WE PROCESSING MULTI AREAS
000252  001264                          BNE     10$                     ; IF NE, YES - KEEP LOOKING
000254  005064  000000                  CLR     A$FLG(R4)               ; END MULTIPLE RESPONSE SEQUENCE
000260  000443                          BR      45$                     ; JOIN COMMON CODE

000262  032764  000002  000000  32$:    BIT     #AF.KNO,A$FLG(R4)       ; PLURAL AREAS ?
000270  001003                          BNE     35$                     ; YES - OK
000272  042764  000001  000000          BIC     #AF.INI,A$FLG(R4)       ; TERMINATE MULTI-RESPONSE SEQUENCE
                                ;
                                ; BUILD RESPONSE MESSAGE
                                ;
000300  112723  177777          35$:    MOVB    #MD.NON,(R3)+           ; NO DETAIL
000304  112723  177777                  MOVB    #MD.NON,(R3)+           ; ...
000310  105023                          CLRB    (R3)+                   ; NO TEXT
000312  105023                          CLRB    (R3)+                   ; NUMERIC FORMAT, LIKE FOR NODE
000314  116423  000004                  MOVB    A$NUM(R4),(R3)+         ; STUFF AREA INTO RESPONSE
000320  105023                          CLRB    (R3)+                   ;  ...

000322  112723  000000                  MOVB    #MP$STA,(R3)+           ; PARAMETER = STATE
000326  112723  000201                  MOVB    #MT$COD!1,(R3)+         ; SINGLE CODED VALUE
000332  112723  000004                  MOVB    #MS$REA,(R3)+           ; REACHABLE

000336  032764  000020  000010          BIT     #MO$STA,A$OPT(R4)       ; INFORMATION TYPE = STATUS ?
000344  001402                          BEQ     40$                     ; IF EQ, NO
000346                                  CALL    GETMHC                  ; GET HOPS AND COST
000352                          40$:    CALL    FMTCIR                  ; FORMAT CIRCUIT ID
000356                                  CALL    NXTNOD                  ; FORMAT NEXT NODE INFO
000362  012700  000001                  MOV     #MS.SUC,R0              ; SUCCESS
000366  000402                          BR      50$                     ; DONE

000370  012700  177600          45$:    MOV     #ME.DON,R0              ; ASSUME WE'RE ALL DONE

000374  166403  000002          50$:    SUB     A$BUF(R4),R3            ; CALCULATE LENGTH OF MESSAGE
000400  010302                          MOV     R3,R2                   ; ...
000402  000241                          CLC                             ; ASSUME SUCCESS
000404  110074  000002                  MOVB    R0,@A$BUF(R4)           ; STUFF RETURN CODE IN RESPONSE
000410  100001                          BPL     55$                     ; IF PL, DONE
000412  000261                          SEC                             ; PASS BACK ERROR

000414                          55$:    RETURN

```

GETEXA - GET EXECUTOR AREA INFORMATION

```
                                                                  L  7
ARREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:08      PAGE 1
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE              REFERENCES

AF.END  = 000040           #5-67      7-207
AF.INI  = 000001           #5-67      6-95       6-154
AF.KNO  = 000002           #5-67      6-100      6-147      6-152
AF.LV2  = 000100           #5-67      7-225      8-276
AF.MUX  = 000020           #5-67      9-361      9-364      9-376
ARREOP    000002 RG        #6-91
ASCIMG  = 000100           #5-62      8-303      8-309      10-419     10-421     10-421
AT$UP   = 000010            6-137     7-217      7-236
A$BUF     000002           #5-67     *6-93       6-179      6-182
A$CIR     000003            6-140     7-220      7-239
A$EXA     000014           #5-67      6-116
A$FLG     000000           #5-67      6-95       6-100      6-147     *6-149      6-152     *6-154      7-207      7-225
                            8-276    *9-361     *9-364      9-376
A$LENX    000027           #5-67
A$MAX     000006           #5-67      6-104
A$NAM     000020           #5-67      11-442
A$NID     000000            6-139     7-219      7-238
A$NUM     000004           #5-67     *6-103      6-104      6-116      6-130      6-162      8-283
A$OPT     000010           #5-67      6-169
A$PAR     000016           #5-67
A$TYP     000002            6-137     7-217      7-236
A$XPT     000012           #5-67      6-110      7-204      8-270      9-340
CODMUL  = 000300           #5-62      10-416
CODSNG  = 000200           #5-62
DECPT   = ******   GX       7-244     11-443
DECSIG  = 000020           #5-62
DECUNS  = 000000           #5-62      8-303      8-309      10-419
D$LNAM    000006            11-446
D$LNUM    000014            7-245     11-444
FMTCIR    001054 R          6-172    #9-329
GETEXA    000416 R          6-118    #7-201
GETMHC    000650 R          6-171    #8-266
HEX     = 000040           #5-62
HEXIMG  = 177777           #5-62
I$$AS   = ******            5-63      8-306
KISAR6  = ******   GX      *6-126    *6-135      7-201     *7-212     *7-228     *7-235     *7-249     *8-287
LF.ACT  = 100000           #5-64
LF.BRO  = 000400           #5-64
LF.BWT  = 000007           #5-64
LF.ENA  = 002000           #5-64
LF.LPB  = 001000           #5-64
LF.MDC  = 000100           #5-64
LF.MFL  = 004000           #5-64
LF.MTP  = 000020           #5-64      9-384
LF.PAC  = 000200           #5-64
LF.RDY  = 040000           #5-64
LF.REA  = 010000           #5-64
LF.SER  = 000040           #5-64
LF.TIM  = 000010           #5-64
LF.UNL  = 020000           #5-64
LF.X2P  = 000000           #5-64
LN.CLO  = 000000           #5-64
```

```
                                                                  M  7
ARREOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:08      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00
```

```
     94                                              .SBTTL  DISPATCH TABLE FOR SAVE ROUTINES
     95
     96                                      ;
     97                                      ; DISPATCH ROUTINE FOR SAVE ROUTINES
     98                                      ;
     99                                      ; NOTE:
    100                                      ;       SAVLOO ALWAYS HAS TO BE THE LAST SAVE ROUTINE CALLED
    101                                      ;       BECAUSE OF THE WAY IT SAVES NODE ADDRESS ENTRIES IN
    102                                      ;       THE CONTEXT AREA.
    103                                      ;
    104 000000                               DISTBL:
    105 000000  000444' 000001                       .WORD   SAVREM, NL.REM          ; SAVE REMOTE NODES
    106 000004  000250' 000002                       .WORD   SAVROL, NL.ROL          ; SAVE ROL NODES
    107 000010  000542' 000004                       .WORD   SAVECL, NL.ECL          ; SAVE ECL NODES
    108 000014  000770' 000010                       .WORD   SAVADJ, NL.ADJ          ; SAVE ADJACENT NODES
    109 000020  001200' 000020                       .WORD   SAVSER, NL.SER          ; SAVE SERVICE NODES
    110 000024  000626' 000040                       .WORD   SAVLOO, NL.LOO          ; SAVE LOOP NODES
    111 000030  000000                               .WORD   0
    112
```

```
                                                                     L 9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-4
Symbol table

MC$NOP   001607      MD$DLV   000020      ME$ARE   000005      MF$SYS   000026      MP$ADS   004406
MC$NOR   000000      MD$DMC   000014      ME$CIR   000003      MF$TES   000022      MP$ALB   000036
MC$NPF   001616      MD$DMF   000046      ME$EXA   000000      MF$TRA   000301      MP$AMC   001640
MC$NPR   001630      MD$DMP   000022      ME$EXE   000200      MF$TRI   000021      MP$AMH   001641
MC$NRE   001200      MD$DMR   000050      ME$LIN   000001      MF$ZER   000025      MP$ANB   004420
MC$NRT   001166      MD$DMV   000042      ME$LOG   000002      ML$ALL   100000      MP$ASB   000024
MC$NVR   001642      MD$DN    000016      ME$MOD   000004      ML$CLS   000000      MP$ASC   004432
MC$P08   000000      MD$DP    000000      ME$NOD   000000      ML$CON   000001      MP$AUS   000512
MC$P11   000001      MD$DPV   000044      ME$NON   177777      ML$EXT   000001      MP$BBT   002165
MC$RFL   002046      MD$DQ    000006      ME$OBJ   000007      ML$FIL   000002      MP$BDF   002164
MC$SBU   002051      MD$DTE   000024      ME$OB2   000004      ML$FIR   000000      MP$BFQ   002121
MC$SFL   002044      MD$DU    000002      ME$PRO   000005      ML$INT   000000      MP$BLK   001616
MC$SYC   000012      MD$DUP   000012      ME$SYS   000006      ML$KNO   140000      MP$BLO   001452
MC$SYL   000036      MD$DV    000026      ME.BLO   177744      ML$MON   000003      MP$BMX   002176
MC$SYR   000050      MD$DZ    000030      ME.CON   177753      ML$SYS   000002      MP$BNP   006330
MC$SYS   000024      MD$FUL   000000      ME.CST   177765      ML$TOP   000001      MP$BRT   001620
MC$SYZ   000000      MD$HAL   000001      ME.DIS   177755      MN$UNL   000377      MP$BSA   002176
MC$TBR   001140      MD$HEL   000002      ME.DON   177600      MO$ACC   000200      MP$BSD   002203
MC$TBS   001141      MD$KCP   000013      ME.FCO   177762      MO$ADD   000002      MP$BSI   002200
MC$TDR   002053      MD$KDP   000034      ME.FIO   177756      MO$ALA   000004      MP$BSP   006331
MC$TIM   000000      MD$KDZ   000036      ME.FOP   177763      MO$ALI   000000      MP$BUF   000170
MC$TYP   007777      MD$KL    000040      ME.FOR   177776      MO$CHA   000040      MP$BUP   006324
MC$UBU   002052      MD$KMX   000054      ME.FUN   177777      MO$CIR   000003      MP$BUS   001643
MC$UFD   002047      MD$KMY   000052      ME.GRO   177745      MO$CLE   000100      MP$CAC   001750
MC$UMR   001132      MD$LEN   000050      ME.HAR   177750      MO$COU   000060      MP$CAS   001762
MC$UMS   001133      MD$PCL   000011      ME.IID   177767      MO$DAC   000000      MP$CAT   002210
MC$VAX   000003      MD$QNA   000005      ME.LCO   177766      MO$DEF   000000      MP$CCS   004406
MC$WID   060000      MD$UNA   000001      ME.LPR   177757      MO$DPR   000000      MP$CHN   002141
MC$WIH   040000      MD$WIT   000002      ME.MPR   177773      MO$ENT   000017      MP$CIR   000144
MC$WIL   020000      MD.ABO   000017      ME.MVE   177771      MO$EVE   000100      MP$CLK   002131
MC$W08   020000      MD.ACC   000010      ME.OPE   177747      MO$INF   000160      MP$CLN   002126
MC$W16   040000      MD.AOB   000016      ME.PLO   177751      MO$INS   000002      MP$CLT   002211
MC$W32   060000      MD.BOB   000011      ME.PMI   177743      MO$LIN   000001      MP$CMB   002142
MC$XBR   001750      MD.DIA   0000[illegible]7      ME.PNA   177752      MO$LOG   000001      MP$CMK   000537
MC$XBS   001751      MD.DOB   000015      ME.PRI   177775      MO$MIR   000031      MP$CMX   002153
MC$XCJ   000322      MD.DUM   000002      ME.PTY   177772      MO$NAM   000001      MP$CND   000310
MC$XCR   002260      MD.FAI   000014      ME.PVA   177760      MO$NIC   000023      MP$CNU   001753
MC$XCS   002261      MD.FNA   000001      ME.RES   177761      MO$NOD   000000      MP$COB   000311
MC$XDR   001762      MD.FOB   000006      ME.ROO   177754      MO$OFF   000001      MP$CON   002126
MC$XDS   001763      MD.LOA   000001      ME.SIZ   177774      MO$ON    000000      MP$COS   001604
MC$XFR   002272      MD.LSH   000020      ME.SYS   177746      MO$OPT   000001      MP$CPF   000230
MC$XFS   002273      MD.NOB   000012      ME.UCO   177770      MO$PER   000200      MP$CPL   000232
MC$XLJ   000323      MD.NON   177777      MF$ACT   177776      MO$PRO   000002      MP$CPT   000226
MC$XLR   002330      MD.PER   000000      MF$ADD   000000      MO$PRS   000001      MP$CPU   000161
MC$XMA   000310      MD.RES   000004      MF$ADJ   177774      MO$REA   000200      MP$CSZ   001755
MC$XMC   002305      MD.ROB   000005      MF$ALL   177775      MO$SET   000000      MP$CUS   002127
MC$XMS   002304      MD.RSH   000013      MF$BYE   000300      MO$STA   000020      MP$CVA   000540
MC$XNR   002332      MD.SDU   000005      MF$CHA   000023      MO$SUM   000000      MP$DAL   002570
MC$XRC   002316      MD.SLO   000003      MF$DUM   000020      MO$TSK   000000      MP$DCO=  001464
MC$XRR   002331      MD.SNA   000000      MF$EVT   000001      MO$VOL   000000      MP$DDT   002177
MC$XRS   002342      MD.TLO   000004      MF$FUP   000264      MO$ZER   000000      MP$DEL   001131
MC$020   000002      MD.UNA   000002      MF$KNO   177777      MP$ACB   000012      MP$DES   000156
MD$CI    000007      MD.UNR   000003      MF$LOA   000017      MP$ACC   000514      MP$DEV   002114
MD$CNA   000003      MD.UOB   000007      MF$LOO   177775      MP$ACT   004526      MP$DFA   001320
MD$COU   000001      MD.VOL   000006      MF$REA   000024      MP$ADD   000766      MP$DHO=  001465
MD$DA    000010      ME$ALI   000010      MF$SIG   177773      MP$ADJ   001440      MP$DIA   000173
MD$DL    000004      ME$AL2   000003      MF$SPF   000302      MP$ADP   006325      MP$DLB   002571
```

```
                                                                     M 9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-5
Symbol table
```

```
	.TITLE	CHKCIR - CHECK A CIRCUIT-ID FORMAT
	.IDENT	/V05.00/
	.ENABL	LC
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CHECK A CIRCUIT-ID FORMAT
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00	08-JUL-81
;	DECNET-11M/S V3.1
;	DECNET-11M-PLUS V1.1
;
; 2.00	16-APR-82
;	DECNET-11M V3.1
;	DECNET-11M-PLUS V1.1
;
; 4.00	07-NOV-83
;	DECNET-11M V4.0
;	DECNET-11M-PLUS V2.0
;
; 5.00	22-JUL-85
;	DECnet-11M/S V4.2
;	DECnet-11M-Plus V3.0
;	DECnet-Micro/RSX V1.0
;
```

Table of contents

```
     5-   55    MACRO CALLS AND LOCAL DATA
     6-   63    BIT DEFINITIONS
     7-   66    CHKLIN - CHECK LINE ID FORMAT
```

```
**FILE**ID**CICHIN




  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     IIIIII     NN      NN
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     IIIIII     NN      NN
CC               II       CC           HH      HH       II       NN      NN
CC               II       CC           HH      HH       II       NN      NN
CC               II       CC           HH      HH       II       NNNN    NN
CC               II       CC           HH      HH       II       NNNN    NN
CC               II       CC           HHHHHHHHHH       II       NN  NN  NN
CC               II       CC           HHHHHHHHHH       II       NN  NN  NN
CC               II       CC           HH      HH       II       NN    NNNN
CC               II       CC           HH      HH       II       NN    NNNN
CC               II       CC           HH      HH       II       NN      NN       ....
CC               II       CC           HH      HH       II       NN      NN       ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     IIIIII     NN      NN       ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     IIIIII     NN      NN       ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10
Table of contents
```

```
    267                                 ;       LISTEN TIMER SUB-EXPRESSION
    268                                 ;
    269 000000                                  FIELD$  SLTM
    270 000000                                   $PARAM  MP$LTM,,PAREND,MP$LTM,+L$PAR
    271 000000                                  FIELD$
    272 000000                                   $NULL   ,,PNAERR
    273                                 ;
    274                                 ;       OWNER SUB-EXPRESSION
    275                                 ;
    276 000000                                  FIELD$  SOWN
    277 000000                                   $PARAM MP$ONR,,PAREND,MP$ONR,+L$PAR
    278 000000                                  FIELD$
    279 000000                                   $BYTE   ME$NOD,NODOWN
    280 000000                                   $BYTE   ME$MOD
    281 000000                                  FIELD$
    282 000000                                   $IMAGE 3,$EXIT,OWNACT
    283 000000                                   $NULL   ,,PVAERR
    284 000000                                  FIELD$  NODOWN
    285 000000                                   $MATCH 2,NODNAM,NODACT
    286 000000                                   $NULL   ,,PVAERR
    287 000000                                  FIELD$  NODNAM
    288 000000                                   $BYTE   0,$EXIT
    289                                 ;
    290                                 ;       TRIBUTARY ADDRESS SUB-EXPRESSION
    291                                 ;
    292 000000                                  FIELD$  STAD
    293 000000                                   $PARAM MP$TRI,,PAREND,MP$TRI,+L$PAR
    294 000000                                  FIELD$
    295 000000                                   $SKIP   ,$EXIT,TADACT
    296                                 ;
    297                                 ;       ACTIVE POLLING RATIO SUB-EXPRESSION
    298                                 ;
    299 000000                                  FIELD$  SACT
    300 000000                                   $PARAM MP$MAC,,PAREND,MP$MAC,+L$PAR
    301 000000                                  FIELD$
    302 000000                                   $SKIP   ,$EXIT,ACTACT
    303                                 ;
    304                                 ;       PROCESS LOCATION SUB-EXPRESSION
    305                                 ;
    306 000000                                  FIELD$  SLOC
    307 000000                                   $PARAM MP$LLO,,PAREND,MP$LLO,+L$PAR
    308 000000                                  FIELD$
    309 000000                                   $BYTE   ML$FIR,$EXIT,,LS.LIN,+L$NTL+LR.STS
    310 000000                                   $BYTE   ML$TOP,$EXIT,,LS.LIN!LS.TOP!LS.OPT,+L$NTL+LR.STS
    311 000000                                   $NULL   ,,PVAERR
    312                                 ;
    313                                 ;       POLL TIMER SUB-EXPRESSION
    314                                 ;
    315 000000                                  FIELD$  SPLT
    316 000000                                   $PARAM MP$SCT,,PAREND,MP$SCT,+L$PAR
    317 000000                                  FIELD$
    318 000000                                   $MATCH 2,$EXIT,PLTACT
    319                                 ;
    320                                 ;       BABBLE TIMER SUB-EXPRESSION
    321                                 ;
    322 000000                                  FIELD$  SBBT
    323 000000                                   $PARAM MP$BBT,,PAREND,MP$SCT,+L$PAR
    772                                               .SBTTL  PARENDD - END OF PARAMETER PARSE
    773                                       ;+
    774                                       ;**-PAREND-END OF PARAMETER PARSE
    775                                       ;
    776                                       ; THIS ROUTINE CLEARS THE CURRENT PARAMETER CELL IN THE
    777                                       ; CONTEXT BLOCK, AND RETURLS$
    778                                       ;
    779                                       ; INPUTS:
    780                                       ;       R4 = ADDRESS OF CONTEXT BLOCK
    781                                       ;
    782                                       ; OUTPUTS:
    783                                       ;       L$PAR(R4) IS CLEARED
    784                                       ;
    785                                       ;-
    786
    787 000536                                PAREND:
    788 000536  005064  000072                        CLR     L$PAR(R4)               ; CLEAR CURRENT PARAMETER CELL
    789 000542                                        RETURN
    790
    791                                        .IF
```

```
                                                .SBTTL  BBTACT - BABBLE TIMER ACTION ROUTINE

                                        ;+
                                        ;
                                        ; BBTACT - BABBLE TIMER ACTION ROUTINE
                                        ;
                                        ; INPUTS:
                                        ;       .MSGPT - POINTER TO 2 BYTE BABBLE TIMER
                                        ;       R4     - POINTER TO CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       L$BBT(R4) - BABBLE TIMER VALUE
                                        ;
                                        ;-
001210                                  BBTACT: CALL    CHKDMP                  ; CHECK FOR DMP
001214  010164  000114                          MOV     R1,L$BBT(R4)            ; SAVE BABBLE TIMER VALUE
001220                                          RETURN
```

```
   1308                                        .ENDC ; .IF NDF R$$11S
   1309
   1310          000001                                .END
```

```
                                                       1
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 8
SAVCTX - SAVE AN ALIAS CONTEXT

                                                .SBTTL  SAVCTX - SAVE AN ALIAS CONTEXT
                                        ;+
                                        ; **-SAVCTX-SAVE AN ALIAS CONTEXT
                                        ;
                                        ; THIS ROUTINE IS CALLED TO SAVE THE NAME AND SCOPE OF AN ALIAS.
                                        ;
                                        ; INPUTS:
                                        ;       R1 = POINTER TO MAPPED ALIAS NAME BLOCK TO BE SAVED
                                        ;       R4 = ADDRESS OF CONTEXT AREA WITH OFFSETS:
                                        ;           -2    = LENGTH OF CONTEXT AREA
                                        ;           A$NUM = NUMBER OF ALIASES ALREADY SAVED
                                        ;       R5 = POINTER INTO CONTEXT AREA FOR SAVED DATA
                                        ;
                                        ; OUTPUTS:
                                        ;       IF C-CLEAR, SUCCESS:
                                        ;           R4 = ADDRESS OF CONTEXT AREA WITH OFFSET:
                                        ;                A$NUM = INCREMENTED BY 1
                                        ;           R5 = NEXT FREE WORD IN CONTEXT AREA
                                        ;       IF C-SET, ALLOCATION FAILURE:
                                        ;           R4 = ADDRESS OF CONTEXT AREA WITH OFFSET:
                                        ;                A$FLG = AF.OVR (BUFFER OVERFLOW)
                                        ;
                                        ; REGISTERS:
                                        ;       R0, R1, R4 ARE PRESERVED
                                        ;
                                        ;-

 000406                                 SAVCTX:
 000406 010402                                  MOV     R4,R2                   ; COPY CONTEXT AREA POINTER
 000410 066402  177776                          ADD     -2(R4),R2               ; POINT TO END OF CONTEXT AREA
 000414 162702  000010                          SUB     #8.,R2                  ; POINT TO BUFFER FOR ''LAST'' ALIAS
 000420 020502                                  CMP     R5,R2                   ; IS THERE ROOM LEFT FOR 1 MORE
 000422 101405                                  BLOS    10$                     ; IF LOS, YES - SAVE IT
 000424 052764  000002  000000                  BIS     #AF.OVR,A$FLG(R4)       ; ELSE, NO - SET OVERFLOW FLAG
 000432 000261                                  SEC                             ; SET ERROR INDICATOR
 000434 000413                                  BR      20$                     ; AND RETURN
 000436 016125  000002                  10$:    MOV     A.NAM(R1),(R5)+         ; SAVE THE ALIAS NAME
 000442 016125  000004                          MOV     A.NAM+2(R1),(R5)+       ; ...
 000446 016125  000006                          MOV     A.NAM+4(R1),(R5)+       ; ...
 000452 016125  000010                          MOV     A.UCB(R1),(R5)+         ; SAVE THE SCOPE OF THE ALIAS
 000456 105264  000010                          INCB    A$NUM(R4)               ; INC SAVED ALIAS COUNT
 000462 000241                                  CLC                             ; SUCCESS !
 000464                                 20$:    RETURN
```

```
                                                       N 1
ALREIN - ALIAS READ INFORMATION MACRO V05.03b Saturday 29-Jun-85 12:05  Page 9
FINDA - FIND A MATCHING ALIAS BLOCK
```

```
     81                                                 .SBTTL  ALREOP - ALIAS READ INFORMATION OPERATE
     82                                         ;+
     83                                         ; **-ALREOP-ALIAS READ INFORMATION OPERATE ROUTINE
     84                                         ;
     85                                         ; THIS ROUTINE IS CALLED TO PREFORM THE READ ALIAS INFORMATION
     86                                         ; OPERATION.
     87                                         ;
     88                                         ; INPUTS:
     89                                         ;       R3 = ADDRESS OF INPUT BUFFER
     90                                         ;       R4 = ADDRESS OF CONTEXT AREA
     91                                         ;
     92                                         ; OUTPUTS:
     93                                         ;       R0 = RETURN CODE
     94                                         ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
     95                                         ;       R3 = ADDRESS OF OUTPUT BUFFER
     96                                         ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
     97                                         ;
     98                                         ; REGISTERS:
     99                                         ;       R3, R4, R5 ARE PRESERVED
    100                                         ;-
    101
    102 000060                                  ALREOP::
    103 000060  004567  000000G                         JSR     R5,$SAVRG               ; SAVE R3-R5
    104 000064  010364  000006                          MOV     R3,A$BUF(R4)            ; SAVE BUFFER POINTER
    105 000070  112723  177600                  10$:    MOVB    #ME.DON,(R3)+           ; ASSUME WE'RE ALL DONE
    106 000074  032764  000001  000000                  BIT     #AF.INI,A$FLG(R4)       ; ARE WE DONE ?
    107 000102  001466                                  BEQ     50$                     ; IF EQ, YES
    108                                         ;
    109                                         ; FORMAT ERROR RETURN IF NO MORE ALIASES TO DISPLAY
    110                                         ;
    111 000104  105364  000010                          DECB    A$NUM(R4)               ; DEC COUNT OF SAVED ALIASES
    112 000110  002022                                  BGE     30$                     ; IF GE, MORE TO DISPLAY
    113 000112  032764  000002  000000                  BIT     #AF.OVR,A$FLG(R4)       ; ELSE, IS THERE AN ''OVERFLOW TEXT''
    114 000120  001457                                  BEQ     50$                     ; IF EQ, NO - JUST RETURN WITH ERROR
    115 000122  112763  177747  177777                  MOVB    #ME.OPE,-1(R3)          ; ELSE, SAY OPERATION FAILED
    116 000130  105023                                  CLRB    (R3)+                   ; ELSE, CLEAR ERROR DETAIL BYTES
    117 000132  105023                                  CLRB    (R3)+                   ; ...
    118
    119                                                 .IF NDF R$$PRO  ; PRO/DECNET
    120
    121 000134  012705  000060                          MOV     #OVRL,R5                ; GET LENGTH OF MESSAGE TEXT
    122 000140  110523                                  MOVB    R5,(R3)+                ; SET SIZE OF MESSAGE TEXT
    123 000142  012700  000000'                         MOV     #OVR,R0                 ; POINT TO MESSAGE
    124 000146  112023                          20$:    MOVB    (R0)+,(R3)+             ; SET MESSAGE INTO BUFFER
    125 000150  005305                                  DEC     R5                      ; LOOP TILL DONE
    126 000152  001375                                  BNE     20$                     ; ...
    127
    128                                                 .ENDC   ; NDF R$$PRO
    129
    130 000154  000441                                  BR      50$                     ; RETURN WITH ERROR
    131                                         ;
    132                                         ; FORMAT INFORMATION FOR NEXT ALIAS
    133                                         ;
    134 000156  112763  000001  177777          30$:    MOVB    #MS.SUC,-1(R3)          ; ASSUME SUCCESSFUL RETURN
    135 000164  105023                                  CLRB    (R3)+                   ; NO ERROR DETAIL
    136 000166  105023                                  CLRB    (R3)+                   ; ...
    137 000170  105023                                  CLRB    (R3)+                   ; NO MESSAGE TEXT
```

M 3

```
**FILE**ID**ALXXIN




  AAAAAA     LL           XX      XX   XX      XX     IIIIII     NN      NN
  AAAAAA     LL           XX      XX   XX      XX     IIIIII     NN      NN
AA      AA   LL           XX      XX   XX      XX       II       NN      NN
AA      AA   LL           XX      XX   XX      XX       II       NN      NN
AA      AA   LL             XX  XX       XX  XX         II       NNNN    NN
AA      AA   LL             XX  XX       XX  XX         II       NNNN    NN
AA      AA   LL               XX           XX           II       NN  NN  NN
AA      AA   LL               XX           XX           II       NN  NN  NN
AAAAAAAAAA   LL             XX  XX       XX  XX         II       NN    NNNN
AAAAAAAAAA   LL             XX  XX       XX  XX         II       NN    NNNN
AA      AA   LL           XX      XX   XX      XX       II       NN      NN       ....
AA      AA   LL           XX      XX   XX      XX       II       NN      NN       ....
AA      AA   LLLLLLLLLL   XX      XX   XX      XX     IIIIII     NN      NN       ....
AA      AA   LLLLLLLLLL   XX      XX   XX      XX     IIIIII     NN      NN       ....



LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

N 3

```
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06
Table of contents
```

```
                                                  M  4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-3
Symbol table

MP$PCT  000144          MP$SWI  002260          MS$UNR  000005          MX$OWN  000040          S2.BEL= 020000
MP$PDT  002114          MP$SYP  002424          MS$VMS  000004          MX$PAR  000036          S2.BRQ= 000020
MP$PHA  000170          MP$TFL  000156          MS.DON  177600          MX$PAS  000010          S2.CR = 002000
MP$PHY  000012          MP$TH1  002206          MS.MOR  000002          MX$RAC  000047          S2.CTO= 040000
MP$PLI  000025          MP$TH2  002205          MS.PAR  000003          MX$RID  000047          S2.CTS= 100000
MP$PLN  002140          MP$TH3  002202          MS.SUC  000001          MX$RPS  000047          S2.ELF= 001000
MP$PLO  000012          MP$TLN  000202          MT$AR4  000003          MX$SID  000040          S2.FLF= 000400
MP$PMC  004374          MP$TLO  000172          MT$ASC  000100          MX$SNK  000377          S2.IRQ= 000200
MP$PNT  002126          MP$TPA  004375          MT$BIL  000001          MX$TYP  000077          S2.OPF= 004000
MP$PRI  004411          MP$TRI  002164          MT$BYS  000011          MX$UID  000020          S2.ORQ= 000100
MP$PRO  002130          MP$TST  000144          MT$CI   000007          M$$CRB= 000124          S2.PCU= 010000
MP$PSS  001763          MP$TYP  002130          MT$CLE  000077          M$$CRX= 000000          S2.RCU= 000001
MP$PST  001762          MP$UCS  004407          MT$COD  000200          M$$FCS= 000000          S2.SRQ= 000040
MP$RET  001631          MP$USR  001750          MT$CON  000001          M$$MGE= 000000          S2.WAL= 000010
MP$RFA  001323          MP$VEC  004410          MT$DMC  000004          M$$NET= 000000          S2.WRA= 000006
MP$RMX  002201          MP$VER  004406          MT$ETH  000006          M$$OVR= 000000          S2.WRB= 000002
MP$RPA  004374          MP$WDF  002165          MT$HEX  000040          NM.CLN  003400          S3.ACR= 000001
MP$RPR  001606          MP$WMX  002177          MT$LPB  000005          NM.INI  002400          S3.CTC= 000004
MP$RRT  002213          MP$XMT  002166          MT$MAX  000037          NM.OPR  003000          S3.FDX= 000200
MP$RST  002212          MP$XPF  000242          MT$MUL  000100          NM.VR2  002401          S3.ICE= 001000
MP$RSV  001754          MP$XPL  000244          MT$NLE  000017          NM.VR3  002402          S3.MHE= 000400
MP$RSZ  001756          MP$XPT  000240          MT$NON  000001          N$$ACC= 000001          S3.NEC= 000020
MP$RTI  001616          MP$XXX  177777          MT$NR4  000005          N$$BUF= 000001          S3.PPT= 020000
MP$RTM  001611          MP$5ST  004406          MT$NTY  000060          N$$LDV= 000001          S3.PTH= 004000
MP$RVE  001604          MS$ACT  000001          MT$OCT  000060          N$$MCP= 000001          S3.RAL= 000010
MP$RVT  000156          MS$ADU  000010          MT$PHA  000002          N$$MLL= 000001          S3.RES= 010000
MP$SAC  000514          MS$ALO  000007          MT$POI  000000          N$$MOV= 000010          S3.RUB= 040000
MP$SAD  000543          MS$ASE  000006          MT$QP2  000010          N$$NCT= 000001          S3.TAB= 000002
MP$SCA  000310          MS$ATR  000011          MT$ROU  000000          N$$PEM= 000001          S3.TME= 002000
MP$SCO  000144          MS$AUT  000000          MT$RO4  000004          P$$P45= 000000          S3.TSY= 000040
MP$SCT  002176          MS$CLE  000003          MT$SEC  000000          P$$WRD= 000000          S3.8BC= 000100
MP$SDU  000203          MS$DED  000004          MT$SGD  000020          Q$$OPT= 000010          S4.ABD= 100000
MP$SDV  000160          MS$DIE  000003          MT$SYS  000002          R$$DER= 000000          S4.ANI= 000400
MP$SEH  004401          MS$DUM  000004          MT$TER  000001          R$$K11= 000001          S4.AVO= 001000
MP$SER  000144          MS$FAI  000013          MT$TRI  000002          R$$SND= 000000          S4.BLK= 002000
MP$SET  000000          MS$GLO  000000          MT$TYP  007777          R$$11M= 000000          S4.DEC= 004000
MP$SGZ  001644          MS$HOL  000002          MT$USD  000000          SAVNAM= ****** GX       S4.DLO= 000100
MP$SID  000176          MS$INA  000002          MT$X25  000003          SF$ANM  000001          S4.EDT= 010000
MP$SIN  000310          MS$LOA  000003          MU$INC  000001          SF$DNM  000002          S4.HFF= 000020
MP$SLI  000156          MS$LOO  000002          MU$OUT  000002          S$$WRG= 000000          S4.HFL= 000007
MP$SLO  000171          MS$DFF  000001          MU$PER  000000          S$$YSZ= 007600          S4.HHT= 000040
MP$SMX  002202          MS$ON   000000          MV$III  000000          S1.DEC= 001000          S4.HSY= 000200
MP$SND  000500          MS$PRO  000006          MV$IV   000001          S1.DPR= 000400          S4.RGS= 020000
MP$SNP  006332          MS$REA  000004          MX$ACT  000020          S1.DSI= 004000          S4.SFC= 040000
MP$SNU  000542          MS$REF  000001          MX$CIR  000020          S1.ESC= 000002          S4.VFL= 000010
MP$SOB  000524          MS$RES  000003          MX$CLN  000020          S1.IBF= 002000          S5.ABP= 100000
MP$SPA  000157          MS$RST  000001          MX$CNM  000006          S1.IBY= 000200          S5.BCC= 020000
MP$SPR  000536          MS$RSX  000002          MX$CON  000006          S1.OBY= 000100          S5.DAO= 040000
MP$SPS  000513          MS$RT   000005          MX$DAC  000020          S1.PTH= 000010          S5.HPC= 000014
MP$STA  000000          MS$SER  000002          MX$DTE  000020          S1.RES= 010000          S5.HPO= 000020
MP$STI  002140          MS$SHU  000002          MX$FIL  000034          S1.RNE= 000020          S5.ITI= 000100
MP$STT  002201          MS$SRV  000007          MX$LGC  000777          S1.RNF= 020000          S5.OXF= 000040
MP$STY  000175          MS$STA  000000          MX$LIN  000020          S1.RSP= 000004          S5.RPO= 002000
MP$SUB  000001          MS$SYN  000012          MX$LON  000006          S1.RST= 000001          S5.SW1= 000001
MP$SUR  000156          MS$TER  000001          MX$NMS  000454          S1.TNE= 040000          S5.TMM= 000002
MP$SUS  000512          MS$TOP  000003          MX$NOD  000006          S1.TSY= 000040          S5.VER= 010000
MP$SVR  000163          MS$TRI  000005          MX$OBJ  000006          S1.USI= 100000          S5.XOF= 000004
```

```
                                                  N  4
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06  Page 8-4
Symbol table
```

```
                                                                              M 5
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 6-4
Symbol table

MD.LSH  000020          ME.SYS  177746          MO$OPT  000001          MP$CPF  000230          MP$LCO  000226
MD.NOB  000012          ME.UCO  177770          MO$PER  000200          MP$CPL  000232          MP$LCT  000156
MD.NON  177777          MF$ACT  177776          MO$PRO  000002          MP$CPT  000226          MP$LHL  000232
MD.PER  000000          MF$ADD  000000          MO$PRS  000001          MP$CPU  000161          MP$LLE  000227
MD.RES  000004          MF$ADJ  177774          MO$REA  000200          MP$CSZ  001755          MP$LLO  004432
MD.ROB  000005          MF$ALL  177775          MO$SET  000000          MP$CUS  002127          MP$LMB  002152
MD.RSH  000013          MF$BYE  000300          MO$STA  000020          MP$CVA  000540          MP$LMX  002200
MD.SDU  000005          MF$CHA  000023          MO$SUM  000000          MP$DAL  002570          MP$LNA  000144
MD.SLO  000003          MF$DUM  000020          MO$TSK  000000          MP$DCO= 001464          MP$LNO  000233
MD.SNA  000000          MF$EVT  000001          MO$VOL  000000          MP$DDT  002177          MP$LOA  000170
MD.TLO  000004          MF$FUP  000264          MO$ZER  000000          MP$DEL  001131          MP$LOG  004514
MD.UNA  000002          MF$KNO  177777          MP$ACB  000012          MP$DES  000156          MP$LOD  000620
MD.UNR  000003          MF$LOA  000017          MP$ACC  000514          MP$DEV  002114          MP$LTM  001613
MD.UOB  000007          MF$LOO  177775          MP$ACT  004526          MP$DFA  001320          MP$LTY  004533
MD.VOL  000006          MF$REA  000024          MP$ADD  000766          MP$DHO= 001465          MP$LWI  000230
ME$ALI  000010          MF$SIG  177773          MP$ADJ  001440          MP$DIA  000173          MP$MAC  004420
ME$AL2  000003          MF$SPF  000302          MP$ADP  006325          MP$DLB  002571          MP$MAD  001630
ME$ARE  000005          MF$SYS  000026          MP$ADS  004406          MP$DLG  004521          MP$MAP  006323
ME$CIR  000003          MF$TES  000022          MP$ALB  000036          MP$DLI= 001466          MP$MAR  001635
ME$EXA  000000          MF$TRA  000301          MP$AMC  001640          MP$DLT  002200          MP$MAV  006321
ME$EXE  000200          MF$TRI  000021          MP$AMH  001641          MP$DRO  001441          MP$MBN  001636
ME$LIN  000001          MF$ZER  000025          MP$ANB  004420          MP$DST  000454          MP$MBR  001637
ME$LOG  000002          ML$ALL  100000          MP$ASB  000024          MP$DTE  002140          MP$MBU  001642
ME$MOD  000004          ML$CLS  000000          MP$ASC  004432          MP$DTY= 001452          MP$MCB  000156
ME$NOD  000000          ML$CON  000001          MP$AUS  000512          MP$DUA  000207          MP$MCO  001632
ME$NON  177777          ML$EXT  000001          MP$BBT  002165          MP$DUC  000210          MP$MDE  004421
ME$OBJ  000007          ML$FIL  000002          MP$BDF  002164          MP$DUM  000202          MP$MHO  001633
ME$OB2  000004          ML$FIR  000000          MP$BFQ  002121          MP$DUP  002127          MP$MLB  000202
ME$PRO  000005          ML$INT  000000          MP$BLK  001616          MP$DVC  002114          MP$MLK  001306
ME$SYS  000006          ML$KNO  140000          MP$BLO  001452          MP$DWE  001321          MP$MLN  001631
ME.BLO  177744          ML$MON  000003          MP$BMX  002176          MP$ELT  000157          MP$MLP  006333
ME.CON  177753          ML$SYS  000002          MP$BNP  006330          MP$ETY  001605          MP$MRB  002171
ME.CST  177765          ML$TOP  000001          MP$BRT  001620          MP$EVE  000311          MP$MRO  001605
ME.DIS  177755          MN$UNL  000377          MP$BSA  002176          MP$FNC  001752          MP$MRP  006322
ME.DON  177600          MO$ACC  000200          MP$BSD  002203          MP$GDT  002222          MP$MRT  002153
ME.FCO  177762          MO$ADD  000002          MP$BSI  002200          MP$GNM  002223          MP$MRV  006320
ME.FIO  177756          MO$ALA  000004          MP$BSP  006331          MP$GRO  002115          MP$MSB  000170
ME.FOP  177763          MO$ALI  000000          MP$BUF  000170          MP$GRP  000541          MP$MVE  000145
ME.FOR  177776          MO$CHA  000040          MP$BUP  006324          MP$GTY  002224          MP$MVI  001634
ME.FUN  177777          MO$CIR  000003          MP$BUS  001643          MP$HAD  001757          MP$MVR  001751
ME.GRO  177745          MO$CLE  000100          MP$CAC  001750          MP$HBT  002142          MP$MWN  002154
ME.HAR  177750          MO$COU  000060          MP$CAS  001762          MP$HDD  000162          MP$MXB  002172
ME.IID  177767          MO$DAC  000000          MP$CAT  002210          MP$HTM  001612          MP$MXC  000466
ME.LCO  177766          MO$DEF  000000          MP$CCS  004406          MP$HWA  002210          MP$MXR  001630
ME.LPR  177757          MO$DPR  000000          MP$CHN  002141          MP$IAT  001322          MP$MXW  002143
ME.MPR  177773          MO$ENT  000017          MP$CIR  000144          MP$IDE  000144          MP$NAA  006335
ME.MVE  177771          MO$EVE  000100          MP$CLK  002131          MP$IDP  006327          MP$NAC  001130
ME.OPE  177747          MO$INF  000160          MP$CLN  002126          MP$IHO  000215          MP$NAP  006320
ME.PLO  177751          MO$INS  000002          MP$CLT  002211          MP$INA  002177          MP$NCT  000240
ME.PMI  177743          MO$LIN  000001          MP$CMB  002142          MP$IND  002204          MP$NET  002114
ME.PNA  177752          MO$LOG  000001          MP$CMK  000537          MP$INI  002201          MP$NLI  000765
ME.PRI  177775          MO$MIR  000031          MP$CMX  002153          MP$ITI  000776          MP$NNA  000764
ME.PTY  177772          MO$NAM  000001          MP$CND  000310          MP$LAA  000231          MP$NOD  000500
ME.PVA  177760          MO$NIC  000023          MP$CNU  001753          MP$LAN  000234          MP$NRA  006334
ME.RES  177761          MO$NOD  000000          MP$COB  000311          MP$LAR  000202          MP$NRB  000214
ME.ROO  177754          MO$OFF  000001          MP$CON  002126          MP$LAS  000012          MP$NSA  001617
ME.SIZ  177774          MO$ON   000000          MP$COS  001604          MP$LBS  000203          MP$NTI  002141
```

```
                                                                              N 5
ARREIN - AREA READ INFORMATION  MACRO V05.03b Saturday 29-Jun-85 12:07  Page 6-5
Symbol table
```

```
                                          .SBTTL  GETEXA - GET EXECUTOR AREA INFORMATION
                                  ;+
                                  ; ** - GETEXA - GET EXECUTOR AREA INFO
                                  ;
                                  ; INPUTS:        R4      = CONTEXT AREA ADDRESS
                                  ;
                                  ; OUTPUT:        R1      = NODE ID (OR 0)
                                  ;                R2      = CHANNEL NUMBER (OR 0)
                                  ;
                                  ; N.B.           ** THIS ROUTINE MUST BE CALLED FROM SYSTEM STATE **
                                  ;-

000416                            GETEXA: SAVMAP                          ;; SAVE APR 6 MAPPING
000422                                    SAVRG   R0                      ;; SAVE REGISTER
000424  005001                            CLR     R1                      ;; NO NODE-ID, YET
000426  016402  000012                    MOV     A$XPT(R4),R2            ;; GET XPT PDV ADDRESS
000432  005762  000000                    TST     Z.DSP(R2)               ;; PROCESS LOADED ?
000436  001477                            BEQ     45$                     ;; IF EQ, NO CHANNEL           ;rjk04
000440  032764  000040  000000            BIT     #AF.END,A$FLG(R4)       ;; IS THIS AN ENDNODE ?
000446  001425                            BEQ     30$                     ;; IF EQ, NO
                                  ;
                                  ; ENDNODE - DESIGNATED ROUTER IS NEXT NODE
                                  ;
000450                                    MAP     Z.DSP(R2)               ;; MAP TO XPT PROCESS
000456  016202  000016                    MOV     Z.DAT(R2),R2            ;; POINT TO XPT DDB
000462  016202  000072                    MOV     N$ADJ1(R2),R2           ;; POINT TO ADJ1
000466  001463                            BEQ     45$                     ;; IF EQ, NONE
000470                                    BIAS    R2                      ;; BIAS ADDRESS FOR APR 6
000500  132762  000010  000002            BITB    #AT$UP,A$TYP(R2)        ;; IS IT UP ?
000506  001453                            BEQ     45$                     ;; IF EQ, NO CHANNEL
000510  016201  000000                    MOV     A$NID(R2),R1            ;; GET DESIGNATED ROUTER NODE-ID
000514  116202  000003                    MOVB    A$CIR(R2),R2            ;; ELSE GET CHANNEL NUMBER
000520  000447                            BR      50$                     ;; GET NODE ID AND RETURN
                                  ;
                                  ; LEVEL 1 ROUTER: RETURN NEAREST AREA ROUTER AND CHANNEL
                                  ;
000522  032764  000100  000000    30$:    BIT     #AF.LV2,A$FLG(R4)       ;; LEVEL 2 ROUTER ?
000530  001036                            BNE     40$                     ;; IF NE, YES
000532  016202  000016                    MOV     Z.DAT(R2),R2            ;; POINT TO XPT DDB
000536                                    MAP     N$ROA1+2(R2)            ;; MAP TO REACHABILITY TABLES
000544  001434                            BEQ     45$                     ;; IF EQ, NO INFORMATION
000546  016200  000026                    MOV     N$ROA1+4(R2),R0         ;; GET ADDRESS OF FIRST ENTRY
000552                                    BIAS    R0                      ;; BIAS IT FOR APR 6
000562  011000                            MOV     (R0),R0                 ;; GET ADDRESS OF ADJACENCY
000564  001424                            BEQ     45$                     ;; IF EQ, NO INFORMATION
000566                                    BIAS    R0                      ;; BIAS IT FOR APR 6
000576                                    MAP     N$ADJ1(R2)              ;; MAP TO ADJACENCIES
000604  132760  000010  000002            BITB    #AT$UP,A$TYP(R0)        ;; IS THIS ONE UP ?
000612  001411                            BEQ     45$                     ;; IF EQ, ALL BETS ARE OFF
000614  016001  000000                    MOV     A$NID(R0),R1            ;; PICK UP AREA ROUTER NODE-ID,
000620  116002  000003                    MOVB    A$CIR(R0),R2            ;; CHANNEL NUMBER OF OUTPUT LINE
000624  000405                            BR      50$                     ;; DONE
                                  ;
                                  ; LEVEL 2 ROUTER: EXEC IS THE NEXT NODE, NO CHANNEL
                                  ;
000626  017701  000000G           40$:    MOV     @DECPT,R1               ;; POINT TO DECNET HOME BLOCK   ; RJK03
000632  016101  000014                    MOV     D$LNUM(R1),R1           ;; GET NEXT NODE
```

```
SYMBOL CROSS REFERENCE                                      CREF   04.00

SYMBOL  VALUE            REFERENCES

LN.DUM  = 000005         #5-64
LN.LOA  = 000004         #5-64
LN.LOO  = 000003         #5-64
LN.OAU  = 000003         #5-64
LN.OFF  = 000001         #5-64
LN.ON   = 000000         #5-64
LN.OOP  = 000004         #5-64
LN.OPE  = 000001         #5-64
LN.REF  = 000002         #5-64
LN.SER  = 000002         #5-64
LN.STA  = 000017         #5-64
LN.SUB  = 000360         #5-64
LN.TRI  = 000006         #5-64
L.COST    000015         #5-64
L.CTL     000012         #5-64        9-373
L.CVA     177776         #5-64
L.DDM     000002         #5-64        9-358
L.DDS     000004         #5-64
L.DLC     000003         #5-64
L.DLM     000006         #5-64
L.DLS     000010         #5-64
L.FLG     000000         #5-64        9-384
L.KRBA    000016         #5-64
L.LEN   = 000022         #5-64
L.MPF     000022         #5-64
L.NMST    000020         #5-64
L.NSTA    000014         #5-64        9-386
L.OWNR    000021         #5-64
L.UNT     000013         #5-64        9-380
MAPADD  = ******   GX     11-455
MD.NON    177777          6-158       6-159
ME.DON    177600          6-177
MO$STA    000020          6-169
MP$DCO  = 001464          8-303       8-303
MP$DHO  = 001465          8-309       8-309
MP$DLI  = 001466          9-333
MP$NXN  = 001476          10-416      10-416
MP$STA    000000          6-165
MS$REA    000004          6-167
MS.SUC    000001          6-174
MT$ASC    000100          5-62        9-337
MT$COD    000200          5-62        5-62        6-166       8-303       8-309       10-416      10-419      10-421
MT$MUL    000100          5-62
MT$NLE    000017          8-303       8-309       10-419
MT$TYP    007777          8-303       8-309       10-416
MXAREA  = 000077         #5-67
M$$MGE  = 000000          6-129       6-136       7-216       7-231       7-234       8-291
NODNAM    001416 R        10-420     #11-439
NXTNOD    001350 R        6-173      #10-414
N$ADJ1    000072          6-135       7-214       7-235
N$MHC1    000036          8-278
N$MHC2    000044          8-282
```

```
SYMBOL CROSS REFERENCE                                      CREF   04.00
```

```
                        .SBTTL  RBNLST - REBUILD NODE LIST IN CONTEXT AREA
                ;+
                ; *** - RBNLST - REBUILD NODE LIST IN CONTEXT AREA
                ;
                ; INPUTS:
                ;       R4 = ADDRESS OF CONTEXT AREA
                ;       R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
                ;
                ; OUTPUTS:
                ;       R4 = ADDRESS OF CONTEXT AREA
                ;       R5 = UPDATED
                ;
                ; NOTE: RBNLST IS CALLED FROM USER STATE.
                ;
                ;-

000032          RBNLST::SWSTK$  10$             ;; GO TO SYSTEM STATE
000036                  SAVMAP                  ;; SAVE APR 6
000042                  CALL    BLNLST          ;; BUILD THE NODE LIST
000046                  RESMAP                  ;; RESTORE APR 6
000052          10$:    RETURN                  ;; RETURN

```

```
                                                                      M 9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-5
Symbol table

MP$DLG  004521      MP$MAP  006323      MP$PCH  002152      MP$SUS  000512      MS$TOP  000003
MP$DLI= 001466      MP$MAR  001635      MP$PCO  000024      MP$SVR  000163      MS$TRI  000005
MP$DLT  002200      MP$MAV  006321      MP$PCT  000144      MP$SWI  002260      MS$UNR  000005
MP$DRO  001441      MP$MBN  001636      MP$PDT  002114      MP$SYP  002424      MS$VMS  000004
MP$DST  000454      MP$MBR  001637      MP$PHA  000170      MP$TFL  000156      MS.DON  177600
MP$DTE  002140      MP$MBU  001642      MP$PHY  000012      MP$TH1  002206      MS.MOR  000002
MP$DTY= 001452      MP$MCB  000156      MP$PLI  000025      MP$TH2  002205      MS.PAR  000003
MP$DUA  000207      MP$MCO  001632      MP$PLN  002140      MP$TH3  002202      MS.SUC  000001
MP$DUC  000210      MP$MDE  004421      MP$PLO  000012      MP$TLN  000202      MT$AR4  000003
MP$DUM  000202      MP$MHO  001633      MP$PMC  004374      MP$TLO  000172      MT$ASC  000100
MP$DUP  002127      MP$MLB  000202      MP$PNT  002126      MP$TPA  004375      MT$BIL  000001
MP$DVC  002114      MP$MLK  001306      MP$PRI  004411      MP$TRI  002164      MT$BYS  000011
MP$DWE  001321      MP$MLN  001631      MP$PRO  002130      MP$TST  000144      MT$CI   000007
MP$ELT  000157      MP$MLP  006333      MP$PSS  001763      MP$TYP  002130      MT$CLE  000077
MP$ETY  001605      MP$MRB  002171      MP$PST  001762      MP$UCS  004407      MT$COD  000200
MP$EVE  000311      MP$MRO  001605      MP$RET  001631      MP$USR  001750      MT$CON  000001
MP$FNC  001752      MP$MRP  006322      MP$RFA  001323      MP$VEC  004410      MT$DMC  000004
MP$GDT  002222      MP$MRT  002153      MP$RMX  002201      MP$VER  004406      MT$ETH  000006
MP$GNM  002223      MP$MRV  006320      MP$RPA  004374      MP$WDF  002165      MT$HEX  000040
MP$GRO  002115      MP$MSB  000170      MP$RPR  001606      MP$WMX  002177      MT$LPB  000005
MP$GRP  000541      MP$MVE  000145      MP$RRT  002213      MP$XMT  002166      MT$MAX  000037
MP$GTY  002224      MP$MVI  001634      MP$RST  002212      MP$XPF  000242      MT$MUL  000100
MP$HAD  001757      MP$MVR  001751      MP$RSV  001754      MP$XPL  000244      MT$NLE  000017
MP$HBT  002142      MP$MWN  002154      MP$RSZ  001756      MP$XPT  000240      MT$NON  000001
MP$HDD  000162      MP$MXB  002172      MP$RTI  001616      MP$XXX  177777      MT$NR4  000005
MP$HTM  001612      MP$MXC  000466      MP$RTM  001611      MP$5ST  004406      MT$NTY  000060
MP$HWA  002210      MP$MXR  001630      MP$RVE  001604      MS$ACT  000001      MT$OCT  000060
MP$IAT  001322      MP$MXW  002143      MP$RVT  000156      MS$ADU  000010      MT$PHA  000002
MP$IDE  000144      MP$NAA  006335      MP$SAC  000514      MS$ALO  000007      MT$POI  000000
MP$IDP  006327      MP$NAC  001130      MP$SAD  000543      MS$ASE  000006      MT$QP2  000010
MP$IHO  000215      MP$NAP  006326      MP$SCA  000310      MS$ATR  000011      MT$ROU  000000
MP$INA  002177      MP$NCT  000240      MP$SCO  000144      MS$AUT  000000      MT$RO4  000004
MP$IND  002204      MP$NET  002114      MP$SCT  002176      MS$CLE  000003      MT$SEC  000000
MP$INI  002201      MP$NLI  000765      MP$SDU  000203      MS$DED  000004      MT$SGD  000020
MP$ITI  000776      MP$NNA  000764      MP$SDV  000160      MS$DIE  000003      MT$SYS  000002
MP$LAA  000231      MP$NOD  000500      MP$SEH  004401      MS$DUM  000004      MT$TER  000001
MP$LAM  000234      MP$NRA  006334      MP$SER  000144      MS$FAI  000013      MT$TRI  000002
MP$LAR  000202      MP$NRF  000214      MP$SET  000000      MS$GLO  000000      MT$TYP  007777
MP$LAS  000012      MP$N[illegible] 001617      MP$SGZ  001644      MS$HOL  000002      MT$USD  000000
MP$LBS  000203      MP$N[illegible] 002141      MP$SID  000176      MS$INA  000002      MT$X25  000003
MP$LCO  000226      MP$NUM  001642      MP$SIN  000310      MS$LOA  000003      MU$INC  000001
MP$LCT  000156      MP$NVE  001274      MP$SLI  000156      MS$LOO  000002      MU$OUT  000002
MP$LHL  000232      MP$NXN= 001476      MP$SLO  000171      MS$OFF  000001      MU$PER  000000
MP$LLE  000227      MP$OAC  000632      MP$SMX  002202      MS$ON   000000      MV$III  000000
MP$LLO  004432      MP$OAN  000620      MP$SND  000500      MS$PRO  000006      MV$IV   000001
MP$LMB  002152      MP$OCO  000776      MP$SNP  006332      MS$REA  000004      MX$ACT  000020
MP$LMX  002200      MP$OHO  000214      MP$SNU  000542      MS$REF  000001      MX$CIR  000020
MP$LNA  000144      MP$ONA  000764      MP$SOB  000524      MS$RES  000003      MX$CLN  000020
MP$LNO  000233      MP$ONR  002114      MP$SPA  000157      MS$RST  000001      MX$CNM  000006
MP$LOA  000170      MP$OQL  001453      MP$SPR  000536      MS$RSX  000002      MX$CON  000006
MP$LOG  004514      MP$OTI  000777      MP$SPS  000513      MS$RT   000005      MX$DAC  000020
MP$LOO  000620      MP$OTY  001022      MP$STA  000000      MS$SER  000002      MX$DTE  000020
MP$LTM  001613      MP$OUS  000777      MP$STI  002140      MS$SHU  000002      MX$FIL  000034
MP$LTY  004533      MP$OVE  001010      MP$STT  002201      MS$SRV  000007      MX$LGC  000777
MP$LWI  000230      MP$OWN  004374      MP$STY  000175      MS$STA  000000      MX$LIN  000020
MP$MAC  004420      MP$PAR  000036      MP$SUB  000001      MS$SYN  000012      MX$LON  000006
MP$MAD  001630      MP$PAS  000513      MP$SUR  000156      MS$TER  000001      MX$NMS  000454
```

```
                                                                      N 9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-6
Symbol table
```

```
     55                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     56                                  ;
     57                                  ; MACRO LIBRARY CALLS
     58                                  ;
     59                                          .MCALL  MANDF$,CIRCX$
     60
     61 000000                                   MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     62
     63 000000                                   CIRCX$  LIST
                                         ;
                                         ; CONTEXT AREA OFFSET DEFINITIONS
                                         ;
        000000                                   .ASECT
                000000                           . = 0
                000020                           L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                         ;
                                         ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                         ; as in the CIRCX$ and LOCCX$ macros.
                                         ;
        000000                           L$NAM:
        000000                           L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                           L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                           L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                           L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                000020                           . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                           L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                           L$TRB:  .BLKB   1                       ; TRIBUTARY NUMBER
        000041                           L$PFG:  .BLKB   1                       ; PARSE FLAGS
        000042                           L$SLT:  .BLKW   1                       ; Current SLT/PVC address
        000044                           L$NXT:  .BLKW   1                       ; Pointer to next SLT/PVC address
        000046                           L$SNM:  .BLKW   1                       ; REMAINING NUMBER OF SYSTEM LINES
        000050                           L$TPT:  .BLKW   1                       ; Current tributary pointer (-1 for PSI)
        000052                           L$TNM:  .BLKB   1                       ; REMAINING NUMBER OF TRIBUTARIES
        000053                           L$CTB:  .BLKB   1                       ; CURRENT TRIBUTARY NUMBER
        000054                           L$CHN:  .BLKB   1                       ; Channel number / X.25 port number and
        000055                           L$PDV:  .BLKB   1                       ; ... PDV assigned
        000056                           L$MSG:  .BLKW   1                       ; ERROR MESSAGE STRING POINTER
        000060                           L$BUF:  .BLKW   1                       ; SAVED BUFFER POINTER
        000062                           L$OPT:  .BLKB   1                       ; SAVED OPTIONS BYTE
        000063                           L$TYP:  .BLKB   1                       ; LINE-ID FORMAT TYPE
        000064                           L$FLG:  .BLKW   1                       ; FLAG WORD
        000066                           L$SLEN: .BLKB   1                       ; Significant length of circuit name
        000067                           L$PRO:                                  ; Line protocol
        000067                           L$MTYP: .BLKB   1                       ; Network management circuit type
        000070                           L$TSZ:                                  ; Block size for transport circuits
        000070                           L$LTM:                                  ; Listen timer for transport circuits
        000070                           L$FLX:  .BLKW   1                       ; Flags word for X.25 circuit commands
        000072                           L$SCR:  .BLKW   7                       ; SCRATCH BUFFER
        000110                           L$ADJ:  .BLKW   1                       ; Pointer to adjacency database
        000112                           L$NOD:  .BLKW   1                       ; Adjacent node address
        000114                           L$COU:  .BLKW   1                       ; Count of adjacency entries
        000116                           L$PLB:  .BLKW   1                       ; PLB address
        000120                           L$PAR:  .BLKW   1                       ; PARAMETER TYPE
        000122                           L$SYL:  .BLKW   1                       ; System line number
        000124                           L$LEN:                                  ; LENGTH OF CONTEXT AREA
        000000                                   .PSECT
                                         ;
                .TITLE  CHKLIN - CHECK A LINE-ID FORMAT
                .IDENT  /V05.00/
                .ENABL  LC
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CHECK A LINE-ID FORMAT
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  14-DEC-79
;       DECNET-11M/S V3.0
;       DECNET-11M-PLUS V1.0
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

Table of contents

```
     5-   53    Macro calls and local data
     6-   70    Context area
     7-   74    Local Symbols
     8-   80    SET CIRCUIT PARSE TABLE
     9-  210    PARSE SUB-EXPRESSIONS
    10-  503    CICHIN - CIRCUIT CHANGE PARAMETERS INITIALIZE
    11-  590    COSACT - LINE COST ACTION ROUTINE
    12-  615    LCTACT - COUNTER TIMER ACTION ROUTINE
    13-  635    OWNACT - CIRCUIT OWNER ACTION ROUTINE
    14-  704    TADACT - TRIBUTARY ADDRESS ACTION ROUTINE
    15-  732    PNAERR - PARAMETER NOT APPLICABLE ERROR RETURN
    15-  734    PVAERR - PARAMETER VALUE ERROR RETURN
    15-  735    CIRERR - CIRCUIT-ID FORMAT ERROR
    16-  772    PAREND - END OF PARAMETER PARSE
    17-  793    ACTACT - MULTIPOINT ACTIVE RATIO ACTION ROUTINE
    18-  818    HTMACT - HELLO TIMER ACTION ROUTINE
    19-  837    CHKDMP - CHECK FOR DMP DEVICE
    20-  871    CHNACT - X.25 PVC Channel Number action routine
    21-  892    CMBACT - X.25 Circuit maximum data action routine
    22-  910    DTEACT - X.25 Circuit DTE action routine
    23-  937    NUMACT - X.25 Circuit DLM number action routine
    24-  963    MXRACT - X.25 Circuit maximum recalls action routine
    25-  980    MXWACT - X.25 Circuit maximum window action routine
    26-  997    RETACT - X.25 Circuit recall timer action routine
    27- 1015    PCKBCD - Pack string of digits in BCD format
    28- 1056    PLTACT - POLL TIMER ACTION ROUTINE
    29- 1074    BBTACT - BABBLE TIMER ACTION ROUTINE
    30- 1092    NMTACT - TRANSMIT TIMER ACTION ROUTINE
    31- 1110    XMTACT - TRANSMIT TIMER ACTION ROUTINE
    32- 1128    BSAACT - ACTIVE BASE ACTION ROUTINE
    33- 1146    BSDACT - DYING BASE ACTION ROUTINE
    34- 1164    BSIACT - INACTIVE BASE ACTION ROUTINE
    35- 1182    INAACT - ACTIVE INCREMENT ACTION ROUTINE
    36- 1200    INDACT - DYING INCREMENT ACTION ROUTINE
    37- 1218    INIACT - INACTIVE INCREMENT ACTION ROUTINE
    38- 1236    TH1ACT - DEAD THRESHOLD ACTION ROUTINE
    39- 1254    TH2ACT - DYING THRESHOLD ACTION ROUTINE
    40- 1272    TH3ACT - INACTIVE THRESHOLD ACTION ROUTINE
    41- 1290    MXBACT - MAXIMUM BLOCKS ACTION ROUTINE
    324 000000                          FIELD$
    325 000000                           $MATCH 2,$EXIT,BBTACT
    326                         ;
    327                         ; NORMAL TIMER SUB-EXPRESSION
    328                         ;
    329 000000                          FIELD$  SNMT
    330 000000                           $PARAM MP$NTI,,PAREND,MP$NTI,+L$PAR
    331 000000                          FIELD$
    332 000000                           $MATCH 2,$EXIT,NMTACT
    333                         ;
    334                         ; TRANSMIT TIMER SUB-EXPRESSION
    335                         ;
    336 000000                          FIELD$  SXMT
    337 000000                           $PARAM MP$XMT,,PAREND,MP$XMT,+L$PAR
    338 000000                          FIELD$
    339 000000                           $MATCH 2,$EXIT,XMTACT
    340                         ;
    341                         ; ACTIVE BASE SUB-EXPRESSION
    342                         ;
    343 000000                          FIELD$  SBSA
    344 000000                           $PARAM MP$BSA,,PAREND,MP$BSA,+L$PAR
    345 000000                          FIELD$
    346 000000                           $MATCH 1,$EXIT,BSAACT
    347                         ;
    348                         ; DYING BASE SUB-EXPRESSION
    349                         ;
    350 000000                          FIELD$  SBSD
    351 000000                           $PARAM MP$BSD,,PAREND,MP$BSD,+L$PAR
    352 000000                          FIELD$
    353 000000                           $MATCH 1,$EXIT,BSDACT
    354                         ;
    355                         ; INACTIVE BASE SUB-EXPRESSION
    356                         ;
    357 000000                          FIELD$  SBSI
    358 000000                           $PARAM MP$BSI,,PAREND,MP$BSI,+L$PAR
    359 000000                          FIELD$
    360 000000                           $MATCH 1,$EXIT,BSIACT
    361                         ;
    362                         ; ACTIVE INCREMENT SUB-EXPRESSION
    363                         ;
    364 000000                          FIELD$  SINA
    365 000000                           $PARAM MP$INA,,PAREND,MP$INA,+L$PAR
    366 000000                          FIELD$
    367 000000                           $MATCH 1,$EXIT,INAACT
    368                         ;
    369                         ; DYING INCREMENT SUB-EXPRESSION
    370                         ;
    371 000000                          FIELD$  SIND
    372 000000                           $PARAM MP$IND,,PAREND,MP$IND,+L$PAR
    373 000000                          FIELD$
    374 000000                           $MATCH 1,$EXIT,INDACT
    375                         ;
    376                         ; INACTIVE INCREMENT SUB-EXPRESSION
    377                         ;
    378 000000                          FIELD$  SINI
    379 000000                           $PARAM MP$INI,,PAREND,MP$INI,+L$PAR
    380 000000                          FIELD$
```

```
                                               .SBTTL  ACTACT - MULTIPOINT ACTIVE RATIO ACTION ROUTINE
                                       ;+
                                       ; **-ACTACT-MULTIPOINT ACTIVE RATIO ACTION ROUTINE
                                       ;
                                       ; THIS ROUTINE CHECKS THAT THE ACTIVE POLLING RATIO SPECIFIED IS WITHIN RANGE,
                                       ; AND SAVES IT IN THE CONTEXT AREA.
                                       ;
                                       ; INPUTS:
                                       ;       .MSGCN  = ACTIVE POLLING RATION
                                       ;       R4      = POINTER TO CONTEXT AREA
                                       ;
                                       ; OUTPUTS:
                                       ;       IF RATIO IS NON ZERO,
                                       ;          L$ACT(R4) = SAVED ACTIVE POLLING RATIO
                                       ;       ELSE, THE PARSE IS TERMINATED
                                       ;-

000544                                 ACTACT:
000544  132764  000002  000041                 BITB    #LP$TRB,L$PFG(R4)       ; TRIBUTARY NUMBER SPECIFIED ?
000552  001743                                 BEQ     PNAERR                  ; IF EQ, NO - PARAMETER NOT APPLICABLE
000554  116701  000000G                        MOVB    .MSGNM,R1               ; GET ACTIVE POLLING RATIO
000560  001743                                 BEQ     PVAERR                  ; IF EQ, PARAMETER VALUE ERROR
000562  110164  000104                         MOVB    R1,L$ACT(R4)            ; ELSE, SAVE IT IN CONTEXT AREA
000566  000763                                 BR      PAREND                  ; END OF PARAMETER PARSE
```

```
                                        .SBTTL  NMTACT - TRANSMIT TIMER ACTION ROUTINE

                                ;+
                                ;
                                ; NMTACT - TRANSMIT TIMER ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 2 BYTE NORMAL TIMER
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$NMT(R4) - NORMAL TIMER VALUE
                                ;
                                ;-
001222                          NMTACT: CALL    CHKDMP                  ; CHECK FOR DMP
001226  010164  000116                  MOV     R1,L$NMT(R4)            ; SAVE NORMAL TIMER VALUE
001232                                  RETURN
```

```
                                                                                   M 16
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-1
Symbol table

ACTACT  000544R         D$$YNM= 000000          LF$REA= 000001          LR.LIN  000002          LS.CX5= 000400
A$$CHK= 000000          END     000450R     002 LF$SEG= 100000          LR.PRM  000030          LS.DLM= 004000
A$$CPS= 000000          E$$XPR= 000000          LF$SKP= 000004          LR.PRO  000002          LS.ECH= 001000
A$$PRI= 000000          FLG$$$= 177777          LF$VR2= 000010          LR.STS  000000          LS.FDX= 004000
A$$TRP= 000000          F$$LVL= 000001          LF$ZER= 000002          LR.TCB  000002          LS.HDX= 010000
BBTACT  001210R         G$$TPP= 000000          LF.ACT= 100000          LR.UNT  000005          LS.LIN= 000003
BSAACT  001246R         G$$TSS= 000000          LF.BRO= 000400          LS$ACT= 000040          LS.LMC= 000007
BSDACT  001260R         G$$TTK= 000000          LF.BWT= 000007          LS$BBT= 010000          LS.NTI= 004000
BSIACT  001272R         G$$WRD= 000000          LF.ENA= 002000          LS$BLK= 001000          LS.ON = 000011
CALL    002024R     002 HTMACT  000570R         LF.LPB= 001000          LS$BSA= 000001          LS.OPT= 000400
CALL1   002030R     002 H.CSP   000000          LF.MDC= 000100          LS$BSD= 000002          LS.PRD= 000002
CF$BLK= 000102          H.CUIC  000010          LF.MFL= 004000          LS$BSI= 000004          LS.PWF= 040000
CHKCIR= ****** GX       H.DSW   000046          LF.MTP= 000020          LS$CHN= 000001          LS.TDP= 002000
CHKDMP  000606R         H.DUIC  000012          LF.PAC= 000200          LS$CMB= 000002          LS.UNF= 020000
CHNACT  000662R         H.EFLM  000004          LF.RDY= 040000          LS$CDS= 000004          LS.X25= 000012
CICHIN  000000RG        H.EFSV  000040          LF.REA= 010000          LS$CUS= 000004          LS.11D= 004000
CIRERR  000476R         H.FCS   000050          LF.SER= 000040          LS$DDT= 001000          LX.CEX= 000004
CLCT    002036R     002 H.FORT  000052          LF.TIM= 000010          LS$DEA= 000100          LX.LIN= 000006
CLECIR  000422R     002 H.FPSA  000042          LF.UNL= 020000          LS$DLM= 001004          LX.PRO= 000005
CMBACT  000710R         H.FPSL= 000062          LF.X2P= 000000          LS$DLT= 002000          L$ACT   000104
COSACT  000170R         H.FPVA  000034          LN.CLD= 000000          LS$DTE= 000010          L$BBT   000114
CRET    002044R     002 H.GARD  000072          LN.DUM= 000005          LS$HBT= 000400          L$BLK   000262
CSTERR  000512R         H.HDLN  000002          LN.LOA= 000004          LS$HTM= 010000          L$BSA   000122
C$DTE   000012          H.IPC   000016          LN.LDD= 000003          LS$INA= 000010          L$BSD   000123
C$FLG   000015          H.IPS   000014          LN.OAU= 000003          LS$IND= 000020          L$BSI   000124
C$GNAM  000016          H.ISP   000020          LN.DFF= 000001          LS$INI= 000040          L$BUF   000060
C$LCN   000010          H.LUN   000076          LN.DN = 000000          LS$LCT= 020000          L$CHN   000054
C$LEN   000016          H.NLUN  000074          LN.DOP= 000004          LS$LMB= 000002          L$CMB   000234
C$LNK   000000          H.NML   000061          LN.DPE= 000001          LS$LDD= 040000          L$COS   000101
C$NAM   000002          H.ODVA  000022          LN.REF= 000002          LS$MDT= 020000          L$CTB   000053
C$NML   000002          H.ODVL  000024          LN.SER= 000002          LS$MRT= 000020          L$CTL   000002
C$PORT  000014          H.DVLY  000054          LN.STA= 000017          LS$MWN= 000040          L$CUS   000260
C$$CKP= 000000          H.PFVA  000032          LN.SUB= 000360          LS$MXB= 001000          L$DDM   000000
C$$ORE= 000400          H.RCVA  000036          LN.TRI= 000006          LS$MXR= 000020          L$DDT   000106
C$$RSH= 177564          H.RRVA  000062          LD.CON  000006          LS$MXW= 000040          L$DEA   000105
DTEACT  000726R         H.SPRI  000060          LD.CSR  000006          LS$NMT= 020000          L$DLT   000110
DV.CCL= 000002          H.TKVA  000026          LD.CTL  000014          LS$NDR= 100000          L$DTE   000240
DV.COM= 020000          H.TKVL  000030          LD.LIN  000004          LS$NTI= 000200          L$DTEL  000236
DV.DIR= 000010          H.VEXT  000056          LD.PAR  000010          LS$NTL= 000001          L$DTEP  000250
DV.EXT= 000400          H.WND   000044          LO.PRI  000010          LS$NUM= 000100          L$FLG   000064
DV.F11= 040000          H.X25   000064          LO.SCR  000014          LS$DWN= 000010          L$FLX   000076
DV.ISP= 002000          INAACT  001304R         LO.STA  000015          LS$PLT= 004000          L$FL1   000066
DV.MBC= 000400          INDACT  001316R         LO.VCT  000012          LS$PRD= 010000          L$FL2   000070
DV.MNT= 100000          INIACT  001330R         LP$MPT= 000010          LS$PVC= 020053          L$HBT   000254
DV.MSD= 000100          I$$RAR= 000000          LP$MUX= 000004          LS$RET= 000200          L$HTM   000226
DV.DSP= 004000          I$$RDN= 000000          LP$NXC= 100000          LS$SER= 100000          L$INA   000125
DV.PSE= 010000          K$$CNT= 177546          LP$TRB= 000002          LS$STA= 000002          L$IND   000126
DV.REC= 000001          K$$CSR= 177546          LP$UNT= 000001          LS$SVC= 000362          L$INI   000127
DV.SDI= 000020          K$$LDC= 000000          LP$WCN= 000040          LS$TAD= 000020          L$LCT   000224
DV.SQD= 000040          K$$TPS= 000074          LP$WDV= 000020          LS$TH1= 000100          L$LEN   000264
DV.SWL= 001000          LCTACT  000212R         LP$WLD= 000360          LS$TH2= 000200          L$LMB   000234
DV.TTY= 000004          LC$EXT  000000          LP$WTR= 000200          LS$TH3= 000400          L$LTM   000230
DV.UMD= 000200          LC$INT  000001          LP$WUN= 000100          LS$XMT= 040000          L$MRT   000253
D$$BUG= 177514          LC$NTL= 000200          LR.AST  000026          LS.AST= 100000          L$MSG   000056
D$$ISK= 000000          LC$DWN= 000400          LR.CIR  000002          LS.CEX= 000001          L$MTYP  000075
D$$L11= 000001          LD$LP = 000000          LR.CTL  000004          LS.CIR= 000010          L$MWN   000252
D$$YNC= 000000          LF$MLT= 040000          LR.EFN  000004          LS.CXD= 010000          L$MXB   000133
```

B 1   J 5   E 10   M 14
C 1   K 5   F 10   N 14

```
                                        .SBTTL  FINDA - FIND A MATCHING ALIAS BLOCK
                                ;+
                                ; **-FINDA-FIND A MATCHING ALIAS BLOCK
                                ;
                                ; THIS ROUTINE FINDS A BLOCK WITH A MATCHING ALIAS NAME AND SCOPE.
                                ;
                                ; INPUTS:
                                ;       R0 = LISTHEAD ADDRESS
                                ;       R4 = POINTER TO CONTEXT AREA WITH:
                                ;            A$ALI = 6 CHAR ALIAS NAME
                                ;            A$SCO = SCOPE OF ALIAS
                                ;
                                ; OUTPUTS:
                                ;       IF CC:
                                ;          R1 = MAPPED ADDRESS OF BLOCK
                                ;       IF CS:
                                ;          NO MATCH IN LIST
                                ;
                                ; REGISTERS:
                                ;       R0, R2, R3, R4, R5 ARE PRESERVED
                                ;-

000466 010001                   FINDA:  MOV     R0,R1                   ; COPY REGISTER TO OUTPUT REGISTER
000470 011146                   10$:    MOV     (R1),-(SP)              ; SET UP FOR MAPPING OF NEXT ANB
000472                                  CALL    @CEACC                  ; MAP TO ANB                  ; RJK03
000476 000261                           SEC                             ; ASSUME FAILURE              ;**-1
000500 012601                           MOV     (SP)+,R1                ; GET MAPPED VIRTUAL ADDRESS
000502 001421                           BEQ     40$                     ; IF EQ, THERE IS NONE (NO MATCH)
000504 026461  000012  000002           CMP     A$ALI(R4),A.NAM(R1)     ; DOES THIS NAME MATCH ?
000512 001366                           BNE     10$                     ; IF NE, NO
000514 026461  000014  000004           CMP     A$ALI+2(R4),A.NAM+2(R1) ; ...
000522 001362                           BNE     10$                     ; ...
000524 026461  000016  000006           CMP     A$ALI+4(R4),A.NAM+4(R1) ; ...
000532 001356                           BNE     10$                     ; ...
000534 026461  000020  000010           CMP     A$SCO(R4),A.UCB(R1)     ; DOES THE SCOPE MATCH ?
000542 001352                           BNE     10$                     ; IF NOT EQ, NO MATCH

000544 000241                           CLC                             ; FOUND IT !
000546                          40$:    RETURN

       000001                           .END
```

```
   138 000172  016401  000022                     MOV     A$NEX(R4),R1           ; POINT TO THE NEXT SAVED CONTEXT
   139 000176  011164  000012                     MOV     (R1),A$ALI(R4)         ; COPY ALIAS NAME INTO BUFFER
   140 000202  016164  000002  000014             MOV     2(R1),A$ALI+2(R4)      ; ...
   141 000210  016164  000004  000016             MOV     4(R1),A$ALI+4(R4)      ; ...
   142 000216                                     CALL    FMTNAM                 ; FORMAT THE ALIAS ID
   143 000222  022164  000020                     CMP     (R1)+,A$SCO(R4)        ; IS THIS THE SAME AS THE LAST ?
   144 000226  010164  000022                     MOV     R1,A$NEX(R4)           ; UPDATE "NEXT" POINTER
   145 000232  001404                             BEQ     40$                    ; IF EQ, SAME SCOPE
   146 000234  014164  000020                     MOV     -(R1),A$SCO(R4)        ; ELSE, SET NEW SCOPE
   147 000240                                     CALL    FMTSCO                 ; AND FORMAT SCOPE PARAMETER
   148 000244                             40$:    CALL    FMTDES                 ; FIND ALIAS BLOCK AND FORMAT DESTINATION
   149 000250  103003                             BCC     50$                    ; IF CC, FOUND IT - RETURN
   150 000252  016403  000006                     MOV     A$BUF(R4),R3           ; ELSE, RESTORE BUFFER POINTER
   151 000256  000704                             BR      10$                    ; AND LOOK FOR NEXT ALIAS
   152                                    ;
   153                                    ; SET UP REGISTERS FOR RETURN
   154                                    ;
   155 000260  010302                     50$:    MOV     R3,R2                  ; SAVE CURRENT POINTER INTO BUFFER
   156 000262  016403  000006                     MOV     A$BUF(R4),R3           ; RESTORE BUFFER POINTER
   157 000266  160302                             SUB     R3,R2                  ; CALC LENGTH OF MESSAGE
   158 000270  111300                             MOVB    (R3),R0                ; SET RETURN CODE
   159 000272  000241                             CLC                            ; ASSUME SUCCESS
   160 000274  100003                             BPL     60$                    ; IF PL, SUCCESS - FINISHED
   161 000276  005064  000000                     CLR     A$FLG(R4)              ; ZERO FLAGS (AF.INI)
   162 000302  000261                             SEC                            ; SET ERROR RETURN
   163 000304                             60$:    RETURN
```

```
ALXXIN - ALIAS INITIALIZATION C MACRO V05.03b Saturday 29-Jun-85 12:06
Table of contents

     5-   53    MACRO CALLS AND LOCAL DATA
     6-   91    CHKALI - CHECK ALIAS FORMAT
     7-  187    CHKSCO - CHECK SCOPE PARAMETER
     8-  325    CHKPAR - CHECK PARAMETER TYPE
```

```
Symbol table

S5.XON= 000010          US.MDE= 000002          U.COTQ  000030          U.TCVP  000043          U2.LWC= 000001
S6.EIO= 000400          US.MDM= 000020          U.CTCB  000026          U.TFLK  000040          U2.L3S= 000004
S6.RDI= 100000          US.MNT= 000100          U.CTL   000004          U.TFRQ  000036          U2.L8S= 010000
S6.RLU= 001000          US.OFL= 000001          U.CTYP  000052          U.TIXL  000060          U2.NEC= 004000
TKTCB = ****** GX       US.PUB= 000004          U.CW1   000010          U.TLPP  000053          U2.PRV= 000010
T$$KMG= 000000          US.PWF= 000010          U.CW2   000012          U.TMTI  000047          U2.RMT= 020000
T$$MIN= 000000          US.RED= 000002          U.CW3   000014          U.TRCK= 000100          U2.R04= 100000
T.UCB = ****** GX       US.SHR= 000001          U.CW4   000016          U.TSTA  000026          U2.SCS= 000004
UC.ALG= 000200          US.SPU= 000002          U.CYL = 000104          U.TST5  000054          U2.SLV= 000200
UC.ATT= 000010          US.UMD= 000010          U.DCB   000000          U.TST6  000056          U2.VT5= 000002
UC.KIL= 000004          US.VV = 000001          U.FCDE= 000042          U.TTAB  000050          U2.7CH= 010000
UC.LGH= 000003          US.WCK= 000010          U.FNUM= 000040          U.TTYP  000046          U3.OPA= 100000
UC.NPR= 000100          UU.ABO= 000400          U.GRP = 000102          U.TUX   000024          U3.PAR= 040000
UC.PWF= 000020          UU.ATN= 000100          U.KCSR= 000032          U.UHVR= 000107          U3.UPC= 020000
UC.QUE= 000040          UU.AVN= 000004          U.KCS6= 000034          U.UIC   000044          U4.CR = 000100
UD.UNS= 000000          UU.GUS= 000010          U.LUIC  177774          U.UNFL= 000052          V$$CTR= 001000
UD.160= 000004          UU.IOS= 002000          U.MEDI= 000040          U.UNIT  000006          W.BATT  000006
UD.200= 000001          UU.ONL= 000020          U.MLUN= 000050          U.UNSZ= 000114          W.BFPD  000014
UD.556= 000002          UU.RCT= 000002          U.MUP   177772          U.UNTI= 000060          W.BHVR  000004
UD.625= 000005          UU.RDY= 000200          U.OWN   177776          U.USVR= 000106          W.BLGH  000020
UD.8K = 000006          UU.SER= 000001          U.RBNS= 000112          U.UTIL= 000036          W.BLPD  000016
UD.800= 000003          UU.SIO= 001000          U.RCTC= 000113          U.VCB = 000034          W.BLVR  000002
UM.CLI= 000036          UU.SPC= 000040          U.RCTS= 000110          U.VSER= 000120          W.BNPD  000015
UM.DSB= 000200          U.ACB   000062          U.RED   000002          U.2MED= 000070          W.BOFF  000012
UM.NBR= 000400          U.ACP = 000032          U.RED2  000034          U2.AT.= 000020          W.BPCB  000000
UM.OVR= 000001          U.ADMA  000066          U.SCB   000020          U2.CRT= 002000          W.BSIZ  000010
US.ABO= 000001          U.AFLG  000064          U.SHST= 000076          U2.DH1= 100000          X$$DBT= 000000
US.BSP= 000002          U.ATT   000022          U.SHUN= 000074          U2.DJ1= 040000          $COTB = ****** GX
US.BSY= 000200          U.BFKT= 000044          U.SPC = 000036          U2.DZ1= 000100          $HEADR= ****** GX
US.FOR= 000040          U.BUF   000024          U.STS   000005          U2.ESC= 001000          $TMLUN= ****** GX
US.FRK= 000002          U.CBF = 000032          U.ST2   000007          U2.HFF= 010000          $$$SYS= 004374
US.KPF= 000001          U.CLI   177772          U.SUB = 000036          U2.HLD= 000040          $$$SYX= 000000
US.LAB= 000004          U.CNT   000030          U.TCHP  000042          U2.LOG= 000400

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000574    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 213
Work  file writes: 148
Size of work file: 28752 Words  ( 113 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:30.26
SY:ALXXIN.V2,[135,134]ALXXIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]ALXXIN
```

```
MP$NUM  001642           MP$SIN  000310           MS$LOA  000003           MU$INC  000001           N$HC2   000056
MP$NVE  001274           MP$SLI  000156           MS$LOO  000002           MU$OUT  000002           N$LV1   000002
MP$NXN= 001476           MP$SLO  000171           MS$OFF  000001           MU$PER  000000           N$LV2   000010
MP$OAC  000632           MP$SMX  002202           MS$ON   000000           MV$III  000000           N$MHC1  000036
MP$OAN  000620           MP$SND  000500           MS$PRO  000006           MV$IV   000001           N$MHC2  000044
MP$OCO  000776           MP$SNP  006332           MS$REA  000004           MXAREA= 000077           N$PLD   000016
MP$OHO  000214           MP$SNU  000542           MS$REF  000001           MX$ACT  000020           N$PRI   000076
MP$ONA  000764           MP$SOB  000524           MS$RES  000003           MX$CIR  000020           N$ROA1  000022
MP$ONR  002114           MP$SPA  000157           MS$RST  000001           MX$CLN  000020           N$ROA2  000030
MP$OQL  001453           MP$SPR  000536           MS$RSX  000002           MX$CNM  000006           N$RTMX  000014
MP$OTI  000777           MP$SPS  000513           MS$RT   000005           MX$CON  000006           N$RTM1  000014
MP$OTY  001022           MP$STA  000000           MS$SER  000002           MX$DAC  000020           N$RTM2  000015
MP$OUS  000777           MP$STI  002140           MS$SHU  000002           MX$DTE  000020           N$RT1   000000
MP$OVE  001010           MP$STT  002201           MS$SRV  000007           MX$FIL  000034           N$RT2   000006
MP$OWN  004374           MP$STY  000175           MS$STA  000000           MX$LGC  000777           N$TCTL  000112
MP$PAR  000036           MP$SUB  000001           MS$SYN  000012           MX$LIN  000020           N$TLC   000100
MP$PAS  000513           MP$SUR  000156           MS$TER  000001           MX$LON  000006           N$TNC   000114
MP$PCH  002152           MP$SUS  000512           MS$TOP  000003           MX$NMS  000454           N$TRC   000106
MP$PCO  000024           MP$SVR  000163           MS$TRI  000005           MX$NOD  000006           N$TTCB  000110
MP$PCT  000144           MP$SWI  002260           MS$UNR  000005           MX$OBJ  000006           N$VER   000066
MP$PDT  002114           MP$SYP  002424           MS$VMS  000004           MX$OWN  000040           N$XLEN  000124
MP$PHA  000170           MP$TFL  000156           MS.DON  177600           MX$PAR  000036           N$$ACC= 000001
MP$PHY  000012           MP$TH1  002206           MS.MOR  000002           MX$PAS  000010           N$$BUF= 000001
MP$PLI  000025           MP$TH2  002205           MS.PAR  000003           MX$RAC  000047           N$$LDV= 000001
MP$PLN  002140           MP$TH3  002202           MS.SUC  000001           MX$RID  000047           N$$MCP= 000001
MP$PLD  000012           MP$TLN  000202           MT$AR4  000003           MX$RPS  000047           N$$MLL= 000001
MP$PMC  004374           MP$TLO  000172           MT$ASC  000100           MX$SID  000040           N$$MOV= 000010
MP$PNT  002126           MP$TPA  004375           MT$BIL  000001           MX$SNK  000377           N$$NCT= 000001
MP$PRI  004411           MP$TRI  002164           MT$BYS  000011           MX$TYP  000077           N$$PEM= 000001
MP$PRO  002130           MP$TST  000144           MT$CI   000007           MX$UID  000020           PDVID  = ****** GX
MP$PSS  001763           MP$TYP  002130           MT$CLE  000077           M$$CRB= 000124           PDVTA  = ****** GX
MP$PST  001762           MP$UCS  004407           MT$COD  000200           M$$CRX= 000000           P$$P45= 000000
MP$RET  001631           MP$USR  001750           MT$CON  000001           M$$FCS= 000000           P$$WRD= 000000
MP$RFA  001323           MP$VEC  004410           MT$DMC  000004           M$$MGE= 000000           Q$$OPT= 000010
MP$RMX  002201           MP$VER  004406           MT$ETH  000006           M$$NET= 000000           R$R0   = ****** GX
MP$RPA  004374           MP$WDF  002165           MT$HEX  000040           M$$OVR= 000000           R$$DER= 000000
MP$RPR  001606           MP$WMX  002177           MT$LPB  000005           NF$ACC= 100000           R$$K11= 000001
MP$RRT  002213           MP$XMT  002166           MT$MAX  000037           NF$EVT= 040000           R$$SND= 000000
MP$RST  002212           MP$XPF  000242           MT$MUL  000100           NF.ACC= 000001           R$$11M= 000000
MP$RSV  001754           MP$XPL  000244           MT$NLE  000017           NF.BCP= 000040           SF$ANM  000001
MP$RSZ  001756           MP$XPT  000240           MT$NON  000001           NF.END= 000200           SF$DNM  000002
MP$RTI  001616           MP$XXX  177777           MT$NR4  000005           NF.EVT= 000002           S$$WRG= 000000
MP$RTM  001611           MP$5ST  004406           MT$NTY  000060           NF.LLI= 000004           S$$YSZ= 007600
MP$RVE  001604           MS$ACT  000001           MT$OCT  000060           NF.LV2= 000100           S1.DEC= 001000
MP$RVT  000156           MS$ADU  000010           MT$PHA  000002           NF.SLI= 000400           S1.DPR= 000400
MP$SAC  000514           MS$ALO  000007           MT$POI  000000           NF.SMC= 000020           S1.DSI= 004000
MP$SAD  000543           MS$ASE  000006           MT$QP2  000010           NF.SWP= 000010           S1.ESC= 000002
MP$SCA  000310           MS$ATR  000011           MT$ROU  000000           NM.CLN  003400           S1.IBF= 002000
MP$SCO  000144           MS$AUT  000000           MT$R04  000004           NM.INI  002400           S1.IBY= 000200
MP$SCT  002176           MS$CLE  000003           MT$SEC  000000           NM.OPR  003000           S1.OBY= 000100
MP$SDU  000203           MS$DED  000004           MT$SGD  000020           NM.VR2  002401           S1.PTH= 000010
MP$SDV  000160           MS$DIE  000003           MT$SYS  000002           NM.VR3  002402           S1.RES= 010000
MP$SEH  004401           MS$DUM  000004           MT$TER  000001           N$ADJ1  000072           S1.RNE= 000020
MP$SER  000144           MS$FAI  000013           MT$TRI  000002           N$ADJ2  000074           S1.RNF= 020000
MP$SET  000000           MS$GLO  000000           MT$TYP  007777           N$CACH  000062           S1.RSP= 000004
MP$SGZ  001644           MS$HOL  000002           MT$USD  000000           N$CRC   000120           S1.RST= 000001
MP$SID  000176           MS$INA  000002           MT$X25  000003           N$HC1   000052           S1.TNE= 040000
```

```
    246 000636  005002              45$:    CLR     R2                      ;; NO CHANNEL
    247
    248 000640                      50$:    RESRG   R0                      ;; RESTORE REGISTER
    249 000642                              RESMAP                          ;; RESTORE APR 6 MAPPING
    250 000646                              RETURN
```

```
                                                                  N  7
ARREOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:08      PAGE 3

SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE            REFERENCES

N$ROA1     000022          7-228     7-230
N$ROA2     000030          6-126     6-128
N$$VCT   = ******          6-126     6-135     6-143     7-201     7-212     7-228     7-235     7-249     8-287
                           8-295
PDVTA    = ******  GX      9-359
R$R1     = ******  GX     *6-120    *6-139    *8-292
R$R2     = ******  GX     *6-121    *6-140
R$$EIS   = ******          8-306
R$$RTR   = ******          9-349
R$$11D   = ******          5-63      8-306
R$$11M   = 000000          5-63      6-129     6-136     7-216     7-231     7-234     8-291
R$$11S   = ******          5-63
SF.ACT   = 0002C0         #5-64
SF.ENA   = 000100         #5-64
SF.LPB   = 000004         #5-64
SF.MFL   = 000040         #5-64
SF.PAC   = 000020         #5-64
SF.REA   = 000010         #5-64
SF.SER   = 000001         #5-64
SF.SVC   = 000002         #5-64
SF.UNL   = 000040         #5-64
SLTMA    = ******  GX      9-346
S$$BAS   = ******          9-349
S.COST     000001         #5-64
S.FLG      000000         #5-64
S.LEN      000004         #5-64
S.NMST     000002         #5-64
S.OWNR     000003         #5-64
TEMP       000000 R       #5-72      8-299     10-417
X$$MCB   = ******          5-63      5-63
ZF.COU   = 001000         #5-63
ZF.DDM   = 000001         #5-63
ZF.DIA   = 004000         #5-63
ZF.DLC   = 000002         #5-63
ZF.DVP   = 100000         #5-63
ZF.INI   = 040000         #5-63
ZF.KMX   = 000020         #5-63
ZF.LLC   = 000004         #5-63
ZF.LMC   = 000100         #5-63
ZF.MAN   = 020000         #5-63
ZF.MFL   = 000010         #5-63
ZF.MTM   = 000400         #5-63
ZF.MUX   = 000040         #5-63      9-362
ZF.PSE   = 002000         #5-63
ZF.SLI   = 010000         #5-63
ZF.TIM   = 000200         #5-63
ZF.X3P   = 000000         #5-63
ZS.ASN   = 100000         #5-63
ZS.BSY   = 140000         #5-63
Z.AVL      000014         #5-63
Z.DAT      000016         #5-63      6-111     7-213     7-227     8-271
Z.DSP      000000         #5-63      5-63      7-205     7-212
```

```
                                                .SBTTL  BLNLST - BUILD NODE LIST IN CONTEXT AREA
                                        ;+
                                        ; *** - BLNLST - BUILD NODE LIST IN CONTEXT AREA
                                        ;
                                        ; INPUTS:
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;       R5 = ADDRESS OF STORAGE FOR NODE ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;       R5 = UPDATED
                                        ;
                                        ; NOTE: BLNLST IS CALLED FROM SYSTEM STATE.
                                        ;
                                        ;-

000054  042764  000002  000000  BLNLST::BIC     #NF.OVR,N$FLAG(R4)              ;; CLEAR CONTEXT AREA OVERFLOW BIT
000062  016464  000050  000046          MOV     N$HCCX(R4),N$HLCX(R4)           ;; MOVE CURRENT TO LAST HIGHEST CONTEXT ADDR
000070  016464  000052  000054          MOV     N$CLOO(R4),N$LLOO(R4)           ;; MOVE CURRENT TO LAST HIGHEST LOOP ADDRESS
000076  012703  000000'                 MOV     #DISTBL,R3                      ;; GET START OF DISPATCH TABLE
000102  012301                  10$:    MOV     (R3)+,R1                        ;; GET ROUTINE ADDRESS
000104  001407                          BEQ     20$                             ;; IF EQ, NO MORE ROUTINES TO CALL
000106  032364  000030                  BIT     (R3)+,N$LFLG(R4)                ;; DISPATCH TO THIS ROUTINE ?
000112  001773                          BEQ     10$                             ;; IF EQ, NO - TRY THE NEXT ONE
000114                                  SAVRG   <R3>                            ;; SAVE TABLE POINTER
000116                                  CALL    (R1)                            ;; ELSE CALL THE ROUTINE
000120                                  RESRG   <R3>                            ;; RESTORE TABLE POINTER
000122  000767                          BR      10$                             ;; CONTINUE

000124  010564  000060          20$:    MOV     R5,N$BFL(R4)                    ;; SAVE BUFFER LIMIT
000130  016403  000050                  MOV     N$HCCX(R4),R3                   ;; GET HIGHEST ADDRESS IN CURRENT CONTEXT
000134  020364  000040                  CMP     R3,N$HADJ(R4)                   ;; HIGHER THAN HIGHEST ADJACENT NODE
000140  103403                          BLO     30$                             ;; IF LO, NO - JUMP
000142  042764  000010  000030          BIC     #NL.ADJ,N$LFLG(R4)              ;; ELSE, CLEAR ADJACENT LIST BIT
000150  020364  000036          30$:    CMP     R3,N$HECL(R4)                   ;; HIGHER THAN HIGHEST ECL NODE
000154  103403                          BLO     40$                             ;; IF LO, NO - JUMP
000156  042764  000004  000030          BIC     #NL.ECL,N$LFLG(R4)              ;; ELSE, CLEAR ECL LIST BIT
000164  020364  000034          40$:    CMP     R3,N$HROL(R4)                   ;; HIGHER THAN HIGHEST ROL NODE
000170  103403                          BLO     50$                             ;; IF LO, NO - JUMP
000172  042764  000002  000030          BIC     #NL.ROL,N$LFLG(R4)              ;; ELSE, CLEAR ROL LIST BIT
000200  020364  000042          50$:    CMP     R3,N$HSER(R4)                   ;; HIGHER THAN HIGHEST SERVICE NODE
000204  103403                          BLO     60$                             ;; IF LO, NO - JUMP
000206  042764  000020  000030          BIC     #NL.SER,N$LFLG(R4)              ;; ELSE, CLEAR SERVICE LIST BIT
000214  020364  000032          60$:    CMP     R3,N$HREM(R4)                   ;; HIGHER THAN HIGHEST REMOTE NODE
000220  103403                          BLO     70$                             ;; IF LO, NO - JUMP
000222  042764  000001  000030          BIC     #NL.REM,N$LFLG(R4)              ;; ELSE, CLEAR REMOTE LIST BIT
000230  026464  000052  000044  70$:    CMP     N$CLOO(R4),N$HLOO(R4)           ;; COMPARE CURRENT HIGH LOOP NODE TO HIGHEST
000236  103403                          BLO     80$                             ;; IF LO, MORE LOOP NODES
000240  042764  000040  000030          BIC     #NL.LOO,N$LFLG(R4)              ;; ELSE, CLEAR LOOP LIST BIT
000246                          80$:    RETURN
```

```
                                                          N 9
BLNLST - BUILD NODE LIST IN CON MACRO V05.03b Saturday 29-Jun-85 12:08  Page 16-6
Symbol table

MX$NOD  000006            N$ALN = 000002            N$PRI   000076            R.ADD   000010            T$LIPS  000006
MX$OBJ  000006            N$AMP   000066            N$QCH   000016            R.FLAG  000012            T$LLD   000012
MX$OWN  000040            N$AMX   000056            N$ROA1  000022            R.LEN   000014            T$LLDC  000045
MX$PAR  000036            N$BFL   000060            N$ROA2  000030            R.LNK   000000            T$LLDL  000012
MX$PAS  000010            N$BUF   000056            N$RTMX  000014            R.NAM   000002            T$LLDO  000012
MX$RAC  000047            N$CACH  000062            N$RTM1  000014            SAVADJ  000770R           T$LLDS  000012
MX$RID  000047            N$CAR   000062            N$RTM2  000015            SAVCTX  001262R           T$LLEN  000046
MX$RPS  000047            N$CHN   000014            N$RT1   000000            SAVECL  000542R           T$LOPR  000002
MX$SID  000040            N$CIR   000034            N$RT2   000006            SAVLOO  000626R           T$LTCL  000024
MX$SNK  000377            N$CLOO  000052            N$SAV   000064            SAVREM  000444R           T$LTIM  000026
MX$TYP  000077            N$CNO   000064            N$SLA   000016            SAVROL  000250R           T$LTPR  000014
MX$UID  000020            N$CRC   000120            N$SNOD  000012            SAVSER  001200R           T$LTPS  000020
M$$CRB= 000124            N$DLA   000020            N$TCTL  000112            SF$ANM  000001            T$NAPL  000004
M$$CRX= 000000            N$DLY   000014            N$TIM   000004            SF$DNM  000002            T$NFE   000000
M$$FCS= 000000            N$ECL   000010            N$TLC   000100            SF.CIR= 000010            T$NLEN  000010
M$$MGE= 000000            N$ELEN  000054            N$TNC   000114            SF.DEV= 000020            T$NNUL  000002
M$$NET= 000000            N$ENC   000042            N$TRC   000106            SF.DIA= 004000            T$NOPL  000006
M$$OVR= 000000            N$ERRC  000022            N$TTCB  000110            SF.DPA= 000100            T$NRNI  000042
NF$BLK= 000100            N$FLAG  000000            N$UCB   000006            SF.DPC= 000200            T$NRPL  000005
NF$DMO= 000010            N$FLG   000005            N$VCB   000010            SF.DUM= 010000            T$NRUL  000007
NF$MOU= 000040            N$FNC   000006            N$VER   000066            SF.HAD= 000004            T$NVR   000001
NF$RST= 000002            N$GENQ  000052            N$XLEN  000124            SF.HST= 000002            T$RPRI  000040
NF$SCN= 000020            N$GTM   000015            N$XPT   000012            SF.LOA= 002000            T$SVC   000034
NF$SHU= 000004            N$HADJ  000040            N$$ACC= 000001            SF.PH3= 020000            T$T5    000030
NF$TIM= 000200            N$HCCX  000050            N$$BUF= 000001            SF.PSS= 000040            T$T6    000032
NF.ARE= 000200            N$HC1   000052            N$$LDV= 000001            SF.SEC= 000400            T$$KMG= 000000
NF.EPM= 002000            N$HC2   000056            N$$MCP= 000001            SF.TER= 001000            T$$MIN= 000000
NF.EXN= 020000            N$HECL  000036            N$$MLL= 000001            S$$WRG= 000000            V$$CTR= 001000
NF.FTT= 040000            N$HIGH  000033            N$$MOV= 000010            S$$YSZ= 007600            W.BATT  000006
NF.INI= 000001            N$HLCX  000046            N$$NCT= 000001            S.ADD   000004            W.BFPD  000014
NF.LPB= 000004            N$HLOO  000044            N$$PEM= 000001            S.CIR   000016            W.BHVR  000004
NF.LV1= 000040            N$HREM  000032            OFFSET= 000002            S.DEV   000020            W.BLGH  000020
NF.LV2= 000100            N$HROL  000034            P$$P45= 000000            S.DIA   000046            W.BLPD  000016
NF.LXX= 004000            N$HSER  000042            P$$WRD= 000000            S.DPA   000032            W.BLVR  000002
NF.MUX= 000010            N$LEN   000270            Q$$OPT= 000010            S.DPC   000036            W.BNPD  000015
NF.OVR= 000002            N$LFLG  000030            RBNLST  000032RG          S.DUM   000050            W.BOFF  000012
NF.PA2= 010000            N$LLOO  000054            RF.INI= 000040            S.FLAG  000002            W.BPCB  000000
NF.REA= 000020            N$LLT   000026            RF.LDA= 000004            S.HAD   000010            W.BSIZ  000010
NF.VR2= 000400            N$LLTM  000024            RF.LOO= 100000            S.HST   000006            X$$DBT= 000000
NF.VR3= 001000            N$LOO   000270            RF.RCR= 000010            S.LENG  000052            ZF.COU= 001000
NF.WLD= 000004            N$LVC   000036            RF.RSV= 000100            S.LNK   000000            ZF.DDM= 000001
NL.ADJ= 000010            N$LV1   000002            RF.RTS= 000020            S.LOA   000044            ZF.DIA= 004000
NL.ECL= 000004            N$LV2   000010            RH.CTF= 000001            S.PSS   000022            ZF.DLC= 000002
NL.LOO= 000040            N$MBXQ  000050            RH.DTF= 000002            S.SEC   000040            ZF.DVP= 100000
NL.REM= 000001            N$MHC1  000036            RH.MSF= 000003            S.TER   000042            ZF.INI= 040000
NL.ROL= 000002            N$MHC2  000044            RL.CKS= 000002            TI.END= 000003            ZF.KMX= 000020
NL.SER= 000020            N$NAM   000020            RL.CTL= 000003            TI.RSV= 000370            ZF.LLC= 000004
NM.CLN  003400            N$NEX   000062            RL.DAH= 000006            TI.RT1= 000002            ZF.LMC= 000100
NM.INI  002400            N$NLEN  000017            RL.DH4= 000025            TI.RT2= 000001            ZF.MAN= 020000
NM.OPR  003000            N$NLN = 000200            RL.SGH= 000004            TI.TYP= 000003            ZF.MFL= 000010
NM.VR2  002401            N$NMP   000070            RL.TYP= 000012            TI.VFY= 000004            ZF.MTM= 000400
NM.VR3  002402            N$NMX   000060            RL.VER= 000006            T$FLAG  000044            ZF.MUX= 000040
N$ACQ   000000            N$NSP   000026            R$R5   = ****** GX        T$LIF   000013            ZF.PSE= 002000
N$ACTL  000032            N$OPT   000002            R$$DER= 000000            T$LIFL  000013            ZF.SLI= 010000
N$ADD   000004            N$PLD   000016            R$$K11= 000001            T$LIFO  000013            ZF.TIM= 000200
N$ADJ1  000072            N$PLLT  000030            R$$SND= 000000            T$LIFS  000013            ZF.X3P= 000000
N$ADJ2  000074            N$PLU   000015            R$$11M= 000000            T$LIN   000000            ZS.ASN= 100000
```

```
                                        ; OFFSETS INTO SCRATCH BUFFER (L$SCR) FOR DMC BASE TABLE COUNTERS
                                        ;
      000124                                  .ASECT
             000000                           .=0
      000000                            B$5:    .BLKB   1                       ; BASE TABLE + 5
      000001                            B$6:    .BLKB   1                       ; BASE TABLE + 6
      000002                            B$7:    .BLKB   1                       ; BASE TABLE + 7
      000003                            B$10:   .BLKB   1                       ; BASE TABLE + 10
      000004                            B$11:   .BLKB   1                       ; BASE TABLE + 11
      000005                            B$12:   .BLKB   1                       ; BASE TABLE + 12
      000006                            B$7S10: .BLKB   1                       ; SUM OF BASE TABLE 7 AND 10
                                                .EVEN
      000000                                    .PSECT
```

```
                                 .SBTTL  MACRO CALLS AND LOCAL DATA
                         ;
                         ; MACRO LIBRARY CALLS
                         ;
                                 .MCALL  MANDF$,CIRCX$

 000000                          MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS

 000000                          CIRCX$  LIST
                         ;
                         ; CONTEXT AREA OFFSET DEFINITIONS
                         ;
 000000                          .ASECT
         000000                   = 0
         000020                  L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                         ;
                         ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                         ; as in the CIRCX$ and LOCCX$ macros.
                         ;
 000000                  L$NAM:
 000000                  L$DDM:  .BLKW   1               ; DEVICE NAME
 000002                  L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
 000003                  L$UNT:  .BLKB   1               ; UNIT NUMBER
 000004                  L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
         000020                  . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
 000020                  L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
 000040                  L$TRB:  .BLKB   1                       ; TRIBUTARY NUMBER
 000041                  L$PFG:  .BLKB   1                       ; PARSE FLAGS
 000042                  L$SLT:  .BLKW   1                       ; Current SLT/PVC address
 000044                  L$NXT:  .BLKW   1                       ; Pointer to next SLT/PVC address
 000046                  L$SNM:  .BLKW   1                       ; REMAINING NUMBER OF SYSTEM LINES
 000050                  L$TPT:  .BLKW   1                       ; Current tributary pointer (-1 for PSI)
 000052                  L$TNM:  .BLKB   1                       ; REMAINING NUMBER OF TRIBUTARIES
 000053                  L$CTB:  .BLKB   1                       ; CURRENT TRIBUTARY NUMBER
 000054                  L$CHN:  .BLKB   1                       ; Channel number / X.25 port number and
 000055                  L$PDV:  .BLKB   1                       ; ... PDV assigned
 000056                  L$MSG:  .BLKW   1                       ; ERROR MESSAGE STRING POINTER
 000060                  L$BUF:  .BLKW   1                       ; SAVED BUFFER POINTER
 000062                  L$OPT:  .BLKB   1                       ; SAVED OPTIONS BYTE
 000063                  L$TYP:  .BLKB   1                       ; LINE-ID FORMAT TYPE
 000064                  L$FLG:  .BLKW   1                       ; FLAG WORD
 000066                  L$SLEN: .BLKB   1                       ; Significant length of circuit name
 000067                  L$PRO:                                  ; Line protocol
 000067                  L$MTYP: .BLKB   1                       ; Network management circuit type
 000070                  L$TSZ:                                  ; Block size for transport circuits
 000070                  L$LTM:                                  ; Listen timer for transport circuits
 000070                  L$FLX:  .BLKW   1                       ; Flags word for X.25 circuit commands
 000072                  L$SCR:  .BLKW   7                       ; SCRATCH BUFFER
 000110                  L$ADJ:  .BLKW   1                       ; Pointer to adjacency database
 000112                  L$NOD:  .BLKW   1                       ; Adjacent node address
 000114                  L$COU:  .BLKW   1                       ; Count of adjacency entries
 000116                  L$PLB:  .BLKW   1                       ; PLB address
 000120                  L$PAR:  .BLKW   1                       ; PARAMETER TYPE
 000122                  L$SYL:  .BLKW   1                       ; System line number
 000124                  L$LEN:                                  ; LENGTH OF CONTEXT AREA
 000000                          .PSECT
                         ;
```

```
                                        .TITLE  CICHIN - CIRCUIT CHANGE PARAMETERS INITIALIZE
                                        .IDENT  /V05.00/
                                        .ENABL  LC
                                ;
                                ; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
                                ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
                                ;
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
                                ; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
                                ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
                                ; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
                                ; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
                                ; TRANSFERRED.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
                                ; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
                                ; CORPORATION.
                                ;
                                ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
                                ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
                                ;
                                ;
                                ; MODULE DESCRIPTION:
                                ;
                                ; NETWORK MANAGEMENT - CIRCUIT CHANGE PARAMETERS INITIALIZE ROUTINE
                                ;
                                ;
                                ; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
                                ;
                                ; IDENT HISTORY:
                                ;
                                ; 1.00  22-JUN-81
                                ;       DECNET-11M/S V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 2.00  16-APR-82
                                ;       DECNET-11M V3.1
                                ;       DECNET-11M-PLUS V1.1
                                ;
                                ; 4.00  07-NOV-83
                                ;       DECNET-11M V4.0
                                ;       DECNET-11M-PLUS V2.0
                                ;
                                ; 5.00  22-JUL-85
                                ;       DECnet-11M/S V4.2
                                ;       DECnet-11M-Plus V3.0
                                ;       DECnet-Micro/RSX V1.0
                                ;
                                ;
```

```
    381 000000                                  $MATCH 1,$EXIT,INIACT
    382                                 ;
    383                                 ; DEAD THRESHOLD SUB-EXPRESSION
    384                                 ;
    385 000000                                 FIELD$  STH1
    386 000000                                  $PARAM MP$TH1,,PAREND,MP$TH1,+L$PAR
    387 000000                                 FIELD$
    388 000000                                  $MATCH 1,$EXIT,TH1ACT
    389                                 ;
    390                                 ; DYING THRESHOLD SUB-EXPRESSION
    391                                 ;
    392 000000                                 FIELD$  STH2
    393 000000                                  $PARAM MP$TH2,,PAREND,MP$TH2,+L$PAR
    394 000000                                 FIELD$
    395 000000                                  $MATCH 1,$EXIT,TH2ACT
    396                                 ;
    397                                 ; INACTIVE THRESHOLD SUB-EXPRESSION
    398                                 ;
    399 000000                                 FIELD$  STH3
    400 000000                                  $PARAM MP$TH3,,PAREND,MP$TH3,+L$PAR
    401 000000                                 FIELD$
    402 000000                                  $MATCH 1,$EXIT,TH3ACT
    403                                 ;
    404                                 ; MAXIMUM BLOCKS SUB-EXPRESSION
    405                                 ;
    406 000000                                 FIELD$  SMXB
    407 000000                                  $PARAM MP$MXB,,PAREND,MP$MXB,+L$PAR
    408 000000                                 FIELD$
    409 000000                                  $MATCH 1,$EXIT,MXBACT
    410                                 ;
    411                                 ; Blocking subexpression
    412                                 ;
    413 000000                                 FIELD$  SBLK
    414 000000                                  $PARAM MP$BLK,,PAREND,MP$BLK,+L$PAR
    415 000000                                 FIELD$
    416 000000                                  $BYTE  MB$ENA,$EXIT,,MB$ENA,+L$BLK
    417 000000                                  $BYTE  MB$DIS,$EXIT,,MB$DIS,+L$BLK
    418 000000                                  $NULL  ,PVAERR
    419                                 ;
    420                                 ; Channel subexpression
    421                                 ;
    422 000000                                 FIELD$  SCHN
    423 000000                                  $PARAM MP$CHN,,PAREND,MP$CHN,+L$PAR
    424 000000                                 FIELD$
    425 000000                                  $MATCH 2,$EXIT,CHNACT
    426                                 ;
    427                                 ; Maximum data subexpression
    428                                 ;
    429 000000                                 FIELD$  SCMB
    430 000000                                  $PARAM MP$CMB,,PAREND,MP$CMB,+L$PAR
    431 000000                                 FIELD$
    432 000000                                  $MATCH 2,$EXIT,CMBACT
    433                                 ;
    434                                 ; Usage subexpression
    435                                 ;
    436 000000                                 FIELD$  SCUS
    437 000000                                  $PARAM MP$CUS,,PAREND,MP$CUS,+L$PAR
    818                                         .SBTTL  HTMACT - HELLO TIMER ACTION ROUTINE
    819
    820                                 ;+
    821                                 ;
    822                                 ; HTMACT - HELLO TIMER ACTION ROUTINE
    823                                 ;
    824                                 ; INPUTS:
    825                                 ;       .MSGPT  = POINTER TO HELLO TIMER
    826                                 ;       R4      = POINTER TO CONTEXT AREA
    827                                 ;
    828                                 ; OUTPUTS:
    829                                 ;          L$HTM(R4) = SAVED HELLO TIMER
    830                                 ;
    831                                 ;-
    832 000570  016701  000000G         HTMACT: MOV     .MSGPT,R1               ; POINT TO HELLO TIMER
    833 000574  112164  000226                  MOVB    (R1)+,L$HTM(R4)         ; SAVE LOW BYTE
    834 000600  111164  000227                  MOVB    (R1),L$HTM+1(R4)        ; AND HIGH BYTE
    835 000604  000754                          BR      PAREND
```

```
                                        .SBTTL  XMTACT - TRANSMIT TIMER ACTION ROUTINE

                                ;+
                                ;
                                ; XMTACT - TRANSMIT TIMER ACTION ROUTINE
                                ;
                                ; INPUTS:
                                ;       .MSGPT - POINTER TO 2 BYTE TRANSMIT TIMER
                                ;       R4     - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       L$XMT(R4) - TRANSMIT TIMER VALUE
                                ;
                                ;-
001234                          XMTACT: CALL    CHKDMP                  ; CHECK FOR DMP
001240  010164  000120                  MOV     R1,L$XMT(R4)            ; SAVE TRANSMIT TIMER VALUE
001244                                  RETURN
```

```
B  1
C  1
D  1
E  1   ALREIN
F  1
G  1
H  1
I  1
J  1
K  1
L  1
M  1
N  1
B  2
C  2
D  2
E  2
F  2
G  2
H  2   ALREOP
I  2
J  2
K  2
L  2
M  2
N  2
B  3
C  3
D  3
E  3
F  3
G  3
H  3
I  3
J  3
K  3
L  3
M  3   ALXXIN
N  3
B  4
C  4
D  4
E  4
F  4
G  4
H  4
I  4
J  4
K  4
L  4
M  4
N  4
B  5
C  5
D  5
E  5   ARREIN
F  5
G  5
H  5
I  5
J  5
K  5
L  5
M  5
N  5
B  6
C  6
D  6
E  6
F  6
G  6   ARREOP
H  6
I  6
J  6
K  6
L  6
M  6
N  6
B  7
C  7
D  7
E  7
F  7
G  7
H  7
I  7
J  7
K  7
L  7
M  7
N  7
B  8
C  8
D  8   BLNLST
E  8
F  8
G  8
H  8
I  8
J  8
K  8
L  8
M  8
N  8
B  9
C  9
D  9
E  9
F  9
G  9
H  9
I  9
J  9
K  9
L  9
M  9
N  9
B 10
C 10
D 10
E 10
F 10
G 10
H 10
I 10
J 10   CHKCIR
K 10
L 10
M 10
N 10
B 11
C 11
D 11
E 11
F 11
G 11
H 11
I 11
J 11
K 11   CHKLIN
L 11
M 11
N 11
B 12
C 12
D 12
E 12
F 12
G 12
H 12
I 12
J 12
K 12
L 12   CICHIN
M 12
N 12
B 13
C 13
D 13
E 13
F 13
G 13
H 13
I 13
J 13
K 13
L 13
M 13
N 13
B 14
C 14
D 14
E 14
F 14
G 14
H 14
I 14
J 14
K 14
L 14
M 14
N 14
B 15
C 15
D 15
E 15
F 15
G 15
H 15
I 15
J 15
K 15
L 15
M 15
N 15
B 16
C 16
D 16
E 16
F 16
G 16
H 16
I 16
J 16
K 16
L 16
M 16
```

```
                                                                                    B 1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-2
Symbol table

L$MXR    000253          MAXCHN=  007777          MC$NOR   000000          MD$DMC   000014          ME$CIR   000003
L$MXW    000252          MB$DIS   000001          MC$NPF   001616          MD$DMF   000046          ME$EXA   000000
L$NAM    000000          MB$ENA   000000          MC$NPR   001630          MD$DMP   000022          ME$EXE   000200
L$NLEN=  000020          MB$FUL   000002          MC$NRE   001200          MD$DMR   000050          ME$LIN   000001
L$NMT    000116          MB$MIX   000002          MC$NRT   001166          MD$DMV   000042          ME$LOG   000002
L$NTI    000256          MB$ONE   000001          MC$NVR   001642          MD$DN    000016          ME$MOD   000004
L$NTL    000134          MB$RXO   000001          MC$P08   000000          MD$DP    000000          ME$NOD   000000
L$NUM    000240          MB$TXO   000000          MC$P11   000001          MD$DPV   000044          ME$NON   177777
L$NUML   000236          MB$ZER   000000          MC$RFL   002046          MD$DQ    000006          ME$OBJ   000007
L$NXT    000044          MC$BAB   006073          MC$SBU   002051          MD$DTE   000024          ME$OB2   000004
L$OPT    000062          MC$BID   001765          MC$SFL   002044          MD$DU    000002          ME$PRO   000005
L$OWN    000102          MC$BMC   001767          MC$SYC   000012          MD$DUP   000012          ME$SYS   000006
L$PAR    000072          MC$BSC   001766          MC$SYL   000036          MD$DV    000026          ME.BLO   177744
L$PDV    000055          MC$CAC   001442          MC$SYR   000050          MD$DZ    000030          ME.CON   177753
L$PFG    000041          MC$CAP   001440          MC$SYS   000024          MD$FUL   000000          ME.CST   177765
L$PLT    000112          MC$CCL   001445          MC$SYZ   000000          MD$HAL   000001          ME.DIS   177755
L$PRO    000075          MC$CDC   002045          MC$TBR   001140          MD$HEL   000002          ME.DON   177600
L$PVC    000004          MC$CDP   001441          MC$TBS   001141          MD$KCP   000013          ME.FCO   177762
L$RET    000256          MC$CIF   001465          MC$TDR   002053          MD$KDP   000034          ME.FIO   177756
L$SCN    000020          MC$CLD   001464          MC$TIM   000000          MD$KDZ   000036          ME.FOP   177763
L$SCR    000166          MC$COU   100000          MC$TYP   007777          MD$KL    000040          ME.FOR   177776
L$SER    000230          MC$CTL   001454          MC$UBU   002052          MD$KMX   000054          ME.FUN   177777
L$SLEN   000074          MC$CTR   001452          MC$UFD   002047          MD$KMY   000052          ME.GRO   177745
L$SLT    000042          MC$CTS   001453          MC$UMR   001132          MD$LEN   000050          ME.HAR   177750
L$SNM    000046          MC$DOV   002050          MC$UMS   001133          MD$PCL   000011          ME.IID   177767
L$STA    000100          MC$LBR   001750          MC$VAX   000003          MD$SNA   000005          ME.LCO   177766
L$TAD    000103          MC$LBS   001751          MC$WID   060000          MD$UNA   000001          ME.LPR   177757
L$TH1    000130          MC$LDI   001774          MC$WIH   040000          MD$WIT   000002          ME.MPR   177773
L$TH2    000131          MC$LDO   001775          MC$WIL   020000          MD.ABO   000017          ME.MVE   177771
L$TH3    000132          MC$LDR   001762          MC$W08   020000          MD.ACC   000010          ME.OPE   177747
L$TNM    000052          MC$LDS   001763          MC$W16   040000          MD.AOB   000016          ME.PLO   177751
L$TPT    000050          MC$LLB   002021          MC$W32   060000          MD.BOB   000011          ME.PMI   177743
L$TRB    000040          MC$LLP   002115          MC$XBR   001750          MD.DIA   000007          ME.PNA   177752
L$TYP    000063          MC$LLR   002007          MC$XBS   001751          MD.DOB   000015          ME.PRI   177775
L$UNT    000003          MC$LOO   000001          MC$XCJ   000322          MD.DUM   000002          ME.PTY   177772
L$XCH    000232          MC$LRB   002020          MC$XCR   002260          MD.FAI   000014          ME.PVA   177760
L$XMT    000120          MC$LRP   002114          MC$XCS   002261          MD.FNA   000001          ME.RES   177761
L$$ASG=  000000          MC$LRR   002006          MC$XDR   001762          MD.FOB   000006          ME.ROO   177754
L$$DRV=  000000          MC$LSI   002032          MC$XDS   001763          MD.LOA   000001          ME.SIZ   177774
L$$P11=  000001          MC$LST   002033          MC$XFR   002272          MD.LSH   000020          ME.SYS   177746
L$$11R=  000000          MC$LUP   006072          MC$XFS   002273          MD.NOB   000012          ME.UCO   177770
L.COST   000015          MC$MAP   010000          MC$XLJ   000323          MD.NON   177777          MF$ACT   177776
L.CTL    000012          MC$MBL   001764          MC$XLR   002330          MD.PER   000000          MF$ADD   000000
L.CVA    177776          MC$MBX   004420          MC$XMA   000310          MD.RES   000004          MF$ADJ   177774
L.DDM    000002          MC$MBY   001752          MC$XMC   002305          MD.ROB   000005          MF$ALL   177775
L.DDS    000004          MC$MPX   004406          MC$XMS   002304          MD.RSH   000013          MF$BYE   000300
L.DLC    000003          MC$NAP   001604          MC$XNR   002332          MD.SDU   000005          MF$CHA   000023
L.DLM    000006          MC$NBR   001130          MC$XRC   002316          MD.SLO   000003          MF$DUM   000020
L.DLS    000010          MC$NBS   001131          MC$XRR   002331          MD.SNA   000000          MF$EVT   000001
L.FLG    000000          MC$NCR   001154          MC$XRS   002342          MD.TLO   000004          MF$FUP   000264
L.KRBA   000016          MC$NCS   001155          MC$020   000002          MD.UNA   000002          MF$KNO   177777
L.LEN =  000022          MC$NML   001274          MD$CI    000007          MD.UNR   000003          MF$LOA   000017
L.MPF    000022          MC$NMR   001142          MD$CNA   000003          MD.UOB   000007          MF$LOO   177775
L.NMST   000020          MC$NMS   001143          MD$COU   000001          MD.VOL   000006          MF$REA   000024
L.NSTA   000014          MC$NNO   001606          MD$DA    000010          ME$AL1   000010          MF$SIG   177773
L.OWNR   000021          MC$NNU   001605          MD$DL    000004          ME$AL2   000003          MF$SPF   000302
L.UNT    000013          MC$NOP   001607          MD$DLV   000020          ME$ARE   000005          MF$SYS   000026
```

```
                                                                                    C 1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-3
Symbol table
```

```
SYMBOL CROSS REFERENCE                                      CREF    04.00

SYMBOL  VALUE               REFERENCES

ZF.X3P  = 000000            #5-65
ZS.ASN  = 100000            #5-65
ZS.BSY  = 140000            #5-65
Z.AVL     000014            #5-65
Z.DAT     000016            #5-65
Z.DSP     000000            #5-65         5-65
Z.FLG     000010            #5-65         13-695
Z.LEN   = 000016            #5-65
Z.LLN     000006            #5-65
Z.MAP     000020            #5-65
Z.NAM     000004            #5-65
Z.PCB     000012            #5-65
Z.SCH     000007            #5-65
$CAT5   = ******  GX         13-676
$EXIT   = 000000            #8-82
$HEADR  = ******  GX         13-697
$RONLY  = ******             8-82
$SAVAL  = ******  GX         13-653       13-662       22-925       23-951
$SAVRG  = ******  GX         10-529
.MSGCN  = ******  GX         19-861
.MSGNM  = ******  GX         11-608       14-721       17-813
.MSGPT  = ******  GX         12-630       13-654       13-663       18-832       19-860       20-885       21-905       22-926       23-952
                             24-976       25-993       26-1010
.MSGST  = ******  GX         15-765
.NPARS  = ******  GX         10-565
```

```
                                         .SBTTL  CICHOP - CIRCUIT CHANGE PARAMETERS OPERATE
                                ;+
                                ; **-CICHOP-CIRCUIT CHANGE PARAMETERS OPERATE ROUTINE
                                ;
                                ; THIS ROUTINE IS CALLED TO PERFORM THE CHANGE CIRCUIT PARAMETERS
                                ; OPERATIONS (SET AND CLEAR).
                                ;
                                ; INPUTS:
                                ;       R3 = ADDRESS OF INPUT/OUTPUT BUFFER
                                ;       R4 = ADDRESS OF CONTEXT AREA (WITH DATA SET UP FROM LICHIN)
                                ;
                                ; OUTPUTS:
                                ;       R0 = NICE RETURN CODE
                                ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                ;       R3 = ADDRESS OF OUTPUT BUFFER
                                ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                ;
                                ; REGISTERS:
                                ;       R3, R4, R5 ARE PRESERVED
                                ;-

000134                          CICHOP::
000134  004567  000000G                 JSR     R5,$SAVRG               ; SAVE R3-R5
000140  010364  000060                  MOV     R3,L$BUF(R4)            ; SAVE BUFFER POINTER
000144  005064  000072                  CLR     L$PAR(R4)               ; CLEAR PARAMETER TYPE
000150  005064  000056                  CLR     L$MSG(R4)               ; CLEAR ERROR MESSAGE POINTER
000154  012700  177600                  MOV     #ME.DON,R0              ; ASSUME WE'RE ALL DONE
000160  016464  000064  000070          MOV     L$FLG(R4),L$FL2(R4)     ; ANY WORK TO DO?
                                        .IF NDF R$$11S
000166  001006                          BNE     10$                     ; BR IF YES
000170  005764  000066                  TST     L$FL1(R4)               ; ANY WORK TO DO?
000174  001003                          BNE     10$                     ; BR if yes

                                        .IF NDF R$$PRO  ; PRO/DECNET

000176  005764  000076                  TST     L$FLX(R4)               ; Any PSI work?

                                        .ENDC   ; NDF R$$PRO

                                        .ENDC   ; NDF R$$11S
000202  001444                          BEQ     20$                     ; BR IF NO
                                ;
                                ; FIND THE NEXT CIRCUIT TO OPERATE ON
                                ;
000204                          10$:
                                        .IF NDF R$$11S & R$$PRO
000204  026427  000050  177777          CMP     L$TPT(R4),#-1           ; Is this a PVC?
000212  001006                          BNE     11$                     ; Branch if not PVC
000214  005764  000042                  TST     L$SLT(R4)               ; Check this is first time through
000220  001003                          BNE     11$                     ; If not, will not be new circuit
000222                                  CALL    $ADPVC                  ; Create new X.25 PVC
000226  103432                          BCS     20$                     ; Exit if error
000230                          11$:
                                        .ENDC
000230                                  CALL    NXTCIR                  ; Find the next circuit
000234  103427                          BCS     20$                     ; If carry, no more circuits - finished
000236                          12$:    CALL    SETUP                   ; Setup for the operation
```

B 4

```
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

CICHOP    000134 RG        #9-151
CICHPX  = ******  GX        9-211
CLENTL  = ******  GX        8-98        12-363
CLNUP     000552 R          10-288     #12-355
DISTBL    000000 R         #8-83        9-197
DISTB1    000056 R         #8-111       9-203
DSPTCH    000440 R          9-198       9-204      #10-277     10-284     10-286
E$NBR     000014           #6-64
E$NBS     000020           #6-64
E$NCR     000034           #6-64
E$NCS     000036           #6-64
E$NIC     000044           #6-64
E$NLEN    000050           #6-64
E$NLLA    000012           #6-64
E$NLNK    000000           #6-64
E$NML     000040           #6-64
E$NMR     000024           #6-64
E$NMS     000030           #6-64
E$NNOD    000002           #6-64
E$NRT     000042           #6-64
E$NRTP    000005           #6-64
E$NSEG    000010           #6-64
E$NTIM    000046           #6-64
E$NUSE    000004           #6-64
E$STRT    000006           #6-64
FMTLIN  = ******  GX        9-254
FNDCTR    000622 RG        #13-390
I$$AS   = ******            6-67
KISAR6  = ******  GX       *13-409     *13-415
LC$NTL  = 000200           #7-73        8-98
LC$OWN  = 000400           #7-73
LF$MLT  = 040000           #7-73
LF$REA  = 000001           #7-73
LF$SEG  = 100000           #7-73
LF$SKP  = 000004           #7-73
LF$VR2  = 000010           #7-73
LF$ZER  = 000002           #7-73
LF.ACT  = 100000           #6-68
LF.BRO  = 000400           #6-68
LF.BWT  = 000007           #6-68
LF.ENA  = 002000           #6-68
LF.LPB  = 001000           #6-68
LF.MDC  = 000100           #6-68
LF.MFL  = 004000           #6-68
LF.MTP  = 000020           #6-68
LF.PAC  = 000200           #6-68
LF.RDY  = 040000           #6-68        11-325
LF.REA  = 010000           #6-68
LF.SER  = 000040           #6-68
LF.TIM  = 000010           #6-68
LF.UNL  = 020000           #6-68
LF.X2P  = 000000           #6-68
```

C 4

```
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 2
```

```
     84                                          .SBTTL  Context area definition
     85
     86 000000                                   CICCX$  LIST
                                          ;
                                          ; CONTEXT AREA OFFSET DEFINITIONS
                                          ;
        000004                                   .ASECT
                       000000                    . = 0
                       000020                    L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                          ;
                                          ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                          ; as in the CIRCX$ and LOCCX$ macros.
                                          ;
        000000                            L$NAM:
        000000                            L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                            L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                            L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                            L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                       000020                     . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                            L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                            L$TRB:  .BLKB   1               ; TRIBUTARY NUMBER
        000041                            L$PFG:  .BLKB   1               ; PARSE FLAGS
        000042                            L$SLT:  .BLKW   1               ; Current SLT/PVC address
        000044                            L$NXT:  .BLKW   1               ; Pointer to next SLT/PVC address
        000046                            L$SNM:  .BLKW   1               ; REMAINING NUMBER OF SYSTEM LINES
        000050                            L$TPT:  .BLKW   1               ; Current tributary pointer (-1 for PSI)
        000052                            L$TNM:  .BLKB   1               ; REMAINING NUMBER OF TRIBUTARIES
        000053                            L$CTB:  .BLKB   1               ; CURRENT TRIBUTARY NUMBER
        000054                            L$CHN:  .BLKB   1               ; Channel number / X.25 port number and
        000055                            L$PDV:  .BLKB   1               ; ... PDV assigned
        000056                            L$MSG:  .BLKW   1               ; ERROR MESSAGE STRING POINTER
        000060                            L$BUF:  .BLKW   1               ; SAVED BUFFER POINTER
        000062                            L$OPT:  .BLKB   1               ; SAVED OPTIONS BYTE
        000063                            L$TYP:  .BLKB   1               ; CIRCUIT-ID FORMAT TYPE
        000064                            L$FLG:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000066                            L$FL1:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000070                            L$FL2:  .BLKW   1               ; FLAG WORD (FOR CURRENT CIRCUIT)
        000072                            L$PAR:  .BLKW   1               ; CURRENT PARAMETER TYPE
        000074                            L$SLEN: .BLKB   1               ; Significant length of circuit name
        000075                            L$PRO:                          ; Line protocol
        000075                            L$MTYP: .BLKB   1               ; Network management circuit type
        000076                            L$FLX:  .BLKW   1               ; Flags word for X.25 circuit commands
        000100                            L$STA:  .BLKB   1               ; CIRCUIT STATE
        000101                            L$COS:  .BLKB   1               ; CIRCUIT COST
        000102                            L$OWN:  .BLKB   1               ; CIRCUIT OWNER (PDV INDEX)
        000103                            L$TAD:  .BLKB   1               ; TRIBUTARY ADDRESS
        000104                            L$ACT:  .BLKB   1               ; MULTIPOINT ACTIVE RATIO
        000105                            L$DEA:  .BLKB   1               ; MULTIPOINT DEAD RATIO
        000106                            L$DDT:  .BLKW   1               ; DMP DEAD TIMER
        000110                            L$DLT:  .BLKW   1               ; DMP DELAY TIMER
        000112                            L$PLT:  .BLKW   1               ; DMP POLL TIMER
        000114                            L$BBT:  .BLKW   1               ; DMP BABBLE TIMER
        000116                            L$NMT:  .BLKW   1               ; DMP NORMAL TIMER
        000120                            L$XMT:  .BLKW   1               ; DMP TRANSMIT TIMER
        000122                            L$BSA:  .BLKB   1               ; DMP ACTIVE BASE
        000123                            L$BSD:  .BLKB   1               ; DMP DYING BASE
        000124                            L$BSI:  .BLKB   1               ; DMP INACTIVE BASE
```

```
001322                          95$:    SOB     R1,90$                  ; Else, continue to check remaining channels
001326                                  RESRG   R5                      ; Restore R5 and
001330  000443                          BR      130$                    ; Ok to turn circuit on

001332  005005                  100$:   CLR     R5                      ; Clear register
001334  156305  177776                  BISB    -2(R3),R5               ; Get system line number
001340  006305                          ASL     R5                      ; Get word offset
001342  067705  000000G                 ADD     @SLTMA,R5               ; Get address of SLT pointer    ; RJK08
001346  011505                          MOV     (R5),R5                 ; Point to SLT
001350  032765  100000  000000          BIT     #LF.ACT,L.FLG(R5)       ; Is line active ?
001356  001761                          BEQ     95$                     ; If EQ, no - continue searching list
001360  032765  000420  000000          BIT     #LF.BRO!LF.MTP,L.FLG(R5); Is this multipoint or broadcast?
001366  001414                          BEQ     105$                    ; BR if yes - error
001370  062705  000022                  ADD     #L.MPF,R5               ; Point to station table
001374  005046                          CLR     -(SP)                   ; Get station number
001376  156316  177777                  BISB    -1(R3),(SP)             ; Get station number
001402  006316                          ASL     (SP)                    ; Multiply station number by 4 (S.LEN)
001404  006316                          ASL     (SP)                    ; ...
001406  062605                          ADD     (SP)+,R5                ; Index into station tables
001410  132765  000200  000000          BITB    #SF.ACT,S.FLG(R5)       ; Is circuit active?
001416  001741                          BEQ     95$                     ; BR if no - continue
001420                          105$:   RESRG   R5                      ; Else, restore R5 and
001422  000442                          BR      170$                    ;  return with error

001424                          110$:
                                        .IF NDF R$$PRO  ; PRO/DECNET


001424  012764  000074' 000056          MOV     #NODMSG,L$MSG(R4)       ; ELSE, SET ERROR MESSAGE

                                        .ENDC   ; NDF R$$PRO

001432  012700  177747          120$:   MOV     #ME.OPE,R0              ; SAY OPERATION FAILED
001436  000434                          BR      170$                    ; AND RETURN

                                        .IF NDF R$$11S
                                ;
                                ; TRY TO ASSIGN LUN TO DLX
                                ;
001440                          130$:   CALL    DLXLUN                  ; ASSIGN A LUN TO NX:
001444  103010                          BCC     140$                    ; IF CC, DLX FOUND, USE QIO TO CHANGE STATE
                                        .ENDC
                                ;
                                ; IF ASSIGN FAILED, NO DLX IN SYSTEM - TRY TO SET STATE BY POKING
                                ;
                                        .IF DF R$$11S
                                130$:
                                        .ENDC
001446                                  SAVRG   <R2>                    ; SAVE R2
001450                                  CALL    TWKLIN                  ; ELSE, "TWEAK" THE STATE OURSELVES
001454                                  RESRG   <R2>                    ; RESTORE R2
001456  103425                          BCS     180$                    ; IF CS, ERROR
001460  116412  000100                  MOVB    L$STA(R4),(R2)          ; SET NEW MANAGEMENT STATE
001464  000422                          BR      180$                    ; AND RETURN

                                        .IF NDF R$$11S
                                ;
                                ; DISPATCH BASED ON FUNCTION REQUESTS
```

```
002770  001457                           BEQ     100$                    ; IF EQ, YES - DONE
002772  116401  000054                   MOVB    L$CHN(R4),R1            ; GET THE CHANNEL NUMBER
002776  116403  000055                   MOVB    L$PDV(R4),R3            ; GET THE PDV ASSIGNED
003002                                   SWSTK$  100$                    ;; ENTER SYSTEM STATE
003006  012702  114224                   MOV     #^RXPT,R2               ;; GET XPT'S PDV
003012                                   CALL    @PDVID                  ;; ...                  ; RJK08
003016  103442                           BCS     70$                     ;; IF CS, NOT THERE - RETURN

003020  120203                           CMPB    R2,R3                   ;; ELSE IS CIRCUIT ASSIGNED TO XPT ?
003022  001040                           BNE     70$                     ;; IF NE, NO - SKIP TASK ACTIVATION

003024  067702  000000G                  ADD     @PDVTA,R2               ;; POINT TO XPT DDB ADDRESS     ; RJK08
003030  011202                           MOV     (R2),R2                 ;; POINT TO XPT DDB
003032  016202  000016                   MOV     Z.DAT(R2),R2            ;; POINT TO DDB
                                 ;
                                 ; LOCATE PLD FOR LINE
                                 ;
003036  016204  000016                   MOV     N$PLD(R2),R4            ;; GET COUNT OF PLDS
003042  016205  000020                   MOV     N$PLD+2(R2),R5          ;; POINT TO ADDRESS OF FIRST ONE

003046  012500                   40$:    MOV     (R5)+,R0                ;; POINT TO NEXT PLB
003050  126001  000004                   CMPB    P$CHN(R0),R1            ;; IS THIS THE CORRECT PLD ?
003054  001403                           BEQ     50$                     ;; IF EQ, FOUND IT
003056                                   SOB     R4,40$                  ;; ELSE KEEP LOOKING
003062  000420                           BR      70$                     ;; RESTORE MAPPING, BACK TO USER STATE
                                 ;
                                 ; SET THE PLD COST BIT AND START THE ROUTING TASK AS NEEDED
                                 ;
003064  152760  000010  000022   50$:    BISB    #PR$LCC,P$STA1(R0)      ;; LEVEL 1 LINE COST CHANGED
003072  152760  000010  000023           BISB    #PR$LCC,P$STA2(R0)      ;; DITTO FOR LEVEL 2
003100  016200  000000                   MOV     N$RT1(R2),R0            ;; POINT TO LEVEL 1 ROUTER TCB
003104  001402                           BEQ     60$                     ;; IF EQ, TASK NOT PRESENT
003106                                   CALL    @EXRQN                  ;; ELSE UNSTOP AND ACTIVATE IT
003112  016200  000006           60$:    MOV     N$RT2(R2),R0            ;; POINT TO LEVEL 2 ROUTER TCB
003116  001402                           BEQ     70$                     ;; IF EQ, NO LEVEL 2 ROUTING TASK
003120                                   CALL    @EXRQN                  ;; ELSE UNSTOP AND ACTIVATE IT
003124                           70$:    RETURN                          ;; BACK TO USER STATE AND RETURN

003126  000261                   90$:    SEC                             ; RETURN FAILURE
003130                           100$:   RETURN
```

```
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 1
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE             REFERENCES

BBTTBL     000120 R        #8-130      22-1018
BSATBL     000156 R        #8-133      22-1027
BSDTBL     000170 R        #8-134      22-1030
BSITBL     000202 R        #8-135      22-1033
CCBGT    = ******   GX      18-742     19-793
CLENTL     000410 RG       #10-198
C.FNC      000010          *18-748    *19-799
C.LIN      000006          *18-745    *19-796
C.NSP      000004          *18-746    *19-797
C.STA      000007          *18-744    *19-795
C.STS      000012          *18-747    *19-798
DEALUN   = ******   GX      13-387
DECPT    = ******   GX      20-888
DLXLUN   = ******   GX      13-468     21-968
DLXQIO   = ******   GX      14-558     14-564     15-626     15-632     16-667     21-993
DMPSET     000424 R         22-1019    22-1022    22-1025    22-1028    22-1031    22-1034    22-1037    22-1040    22-1043
                            22-1046    22-1049    22-1052    22-1055   #23-1076
D$NN       000060           20-889
EXRQN    = ******   GX      20-922     20-925
E$NBR      000014          #5-68
E$NBS      000020          #5-68
E$NCR      000034          #5-68
E$NCS      000036          #5-68
E$NIC      000044          #5-68
E$NLEN     000050          #5-68
E$NLLA     000012          #5-68
E$NLNK     000030          #5-68
E$NML      000040          #5-68
E$NMR      000024          #5-68
E$NMS      000030          #5-68
E$NNOD     000002          #5-68
E$NRT      000042          #5-68
E$NRTP     000005          #5-68
E$NSEG     000010          #5-68
E$NTIM     000046          #5-68
E$NUSE     000004          #5-68
E$STRT     000006          #5-68
FC.CTL   = 000006           18-748     19-799
FNDCTR   = ******   GX      12-299
FS.STP   = 002000           19-799
FS.STR   = 001000           18-748
HEADR    = ******   GX      13-403
H.LUN      000076           13-404
INATBL     000214 R        #8-136      22-1036
INDTBL     000226 R        #8-137      22-1039
INITBL     000240 R        #8-138      22-1042
INSMSG     000000 R        #7-98       11-244
I$$AS    = ******            5-72
KISAR6   = ******   GX      12-298    *12-304
LC$NTL   = 000200          #6-86
LC$OWN   = 000400          #6-86
LF$MLT   = 040000          #6-86
```

```
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 2
```

```
    53                                          .SBTTL  MACRO CALLS AND LOCAL DATA
    54                                  ;
    55                                  ; MACRO LIBRARY CALLS
    56                                  ;
    57                                          .MCALL  SAVRG,RESRG
    58                                  ;       .MCALL  BFFDF$,CDBDF$
    59                                          .MCALL  CEACC$,CTRDF$,MANDF$,PHBDF$
    60                                          .MCALL  PLBDF$,PVCDF$,SLTDF$,SVCDF$,XCBDF$
    61                                          .MCALL  MAP,BIAS,RETC,SAVMAP,RESMAP
    62                                          .MCALL  CICCX$,CALL,CALLR,CALLX
    63
    64
    65                                  ;       BFFDF$                          ; PSV buffer definitions
    66                                  ;       CDBDF$                          ; PSV Circuit Database definitions
    67 000000                                   CTRDF$                          ; Define counter block offsets
    68 000000                                   MANDF$                          ; Define network management symbols
    69 000000                                   PHBDF$                          ; Define PSI home block offsets
    70 000000                                   PLBDF$                          ; Define PLB symbols (DLM flags)
    71 000000                                   PVCDF$                          ; Define PVC offsets and symbols
    72 000000                                   SLTDF$                          ; Define SLT offsets
    73 000000                                   SVCDF$                          ; Define SVC offsets
    74 000000                                   XCBDF$                          ; Define XCB offsets and symbols
    75
    76                                          .MACRO  PARMAP  FLAG,SOURCE,SLEN,SCALC,CBOFF,CBLEN,CBCALC
    77                                          .IF NB  FLAG
    78                                          .WORD   FLAG
    79                                          .WORD   SLEN,CBLEN
    80                                          .WORD   SCALC,CBCALC,SOURCE,CBOFF
    81                                          .IFF
    82                                          .WORD   -1
    83                                          .ENDC
    84                                          .ENDM
    85
    86                                  ;
    87                                  ; Buffer allocation overhead (for $XLDBGT, $XLDBRT routines)
    88                                  ;
    89          000000                          $XLBOV = 0
    90
    91                                  ;
    92                                  ; Offsets for registers saved by $SAVAL
    93                                  ;
    94          000002                          R$R0    = 2
    95          000004                          R$R1    = 4
    96          000006                          R$R2    = 6
    97          000010                          R$R3    = 10
    98          000012                          R$R4    = 12
    99          000014                          R$R5    = 14
   100          000016                          R$PC    = 16
```

```
                                                .SBTTL  SETLCT - SET COUNTER TIMER
                                        ;+
                                        ;
                                        ; SETLCT - SET COUNTER TIMER
                                        ; CLELCT - CLEAR COUNTER TIMER
                                        ;
                                        ; INPUTS:
                                        ;       R4 - ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       COUNTER TIMER UPDATED
                                        ;
                                        ;-
001070  022764  177777  000050          SETLCT: CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
001076  001042                                  BNE     10$                     ; BR IF NO
001100  000241                                  CLC                             ; INDICATE SUCCESS

001102                                          SWSTK$  20$                     ;; ENTER SYSTEM STATE
001106                                          SAVMAP                          ;; SAVE CURRENT MAPPING
001112                                          CEACC$  L$SLT(R4),R0            ;; GET MAPPED ADDRESS OF PVC NAME BLOCK
001124  005001                                  CLR     R1
001126  156001  000014                          BISB    C$PORT(R0),R1           ;; Get port number (NO sign extension)
001132  010100                                  MOV     R1,R0
001134  017701  000000G                         MOV     @PSIPT,R1               ;; GET ADDRESS OF PSI HOME BLOCK; RJK01
001140  016101  000020                          MOV     H$PTB(R1),R1            ;; GET ADDRESS OF PORT TABLE    ;**-1
001144  006300                                  ASL     R0                      ;; CONVERT TO WORD INDEX
001146  060001                                  ADD     R0,R1                   ;; INDEX INTO PORT TABLE
001150                                          CEACC$  R1                      ;; GET MAPPED ADDRESS OF ENTRY
001160                                          CEACC$  (R1),R1                 ;; GET MAPPED ADDRESS OF XCB
001170  016461  000224  000060                  MOV     L$LCT(R4),X$CTIM(R1)    ;; STORE COUNTER TIMER VALUE
001176                                          RESMAP                          ;; RESTORE PREVIOUS MAPPING
001202                                          RETURN

001204  012700  177752                  10$:    MOV     #ME.PNA,R0              ; INDICATE INVALID PARAMETER
001210  012764  000156  000072                  MOV     #MP$LCT,L$PAR(R4)       ; SET PARAMETER NUMBER
001216  000261                                  SEC                             ; INDICATE ERROR
001220                                  20$:    RETURN
```

```
                                                                      B 11
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 7
MACRO CROSS REFERENCE                                   CREF    04.00
MACRO NAME          REFERENCES

BIAS                #5-61
CALL                #5-62     8-152     9-195     10-217    10-220    10-223    10-225    10-228    10-231    14-361
                      14-363  15-390    15-392    15-400    15-401    16-419
CALLR               #5-62     8-150     8-162
CALLX               #5-62
CEACC$              #5-59     15-392    15-400    15-401
CICCX$              #5-62     6-104
CTRDF$              #5-59     5-67
MANDF$              #5-59     5-68
MAP                 #5-61     10-227
PARMAP              #5-76     7-123     7-124     7-125     7-126     7-127     7-128     7-129     7-130     7-131
                      7-132
PHBDF$              #5-59     5-69
PLBDF$              #5-60     5-70
PVCDF$              #5-60     5-71
RESMAP              #5-61     10-238    14-370    15-403
RESRG               #5-57
RETC                #5-61     14-369
RETURN                8-174   9-196     9-200     10-239    10-244    11-298    12-319    13-346    14-371    15-404
                      15-409  16-421
SAVMAP              #5-61     10-218    14-362    15-391
SAVRG               #5-57
SLTDF$              #5-60     5-72
SVCDF$              #5-60     5-73
SWSTK$                10-217  14-361    15-390    16-419
XCBDF$              #5-60     5-74
```

```
                                                                      C 11
**FILE**ID**CIRECH
```

```
                                .SBTTL  FNDPLD - Find physical link data base
                        ;+
                        ; **-FNDPLD-FIND PHYSICAL LINK DATA BASE FOR A LINE
                        ;
                        ; INPUTS:
                        ;       R4 = ADDRESS OF CONTEXT BLOCK
                        ;
                        ; OUTPUTS:
                        ;       R2 = ADDRESS OF PLD FOR THE LINE
                        ;       IF CS, LINE IN NOT OWNED BY TRANSPORT
                        ;
                        ;-

001064                  FNDPLD:
001064                          SWSTK$  20$                     ;; ENTER SYSTEM STATE
001070  012702  114224          MOV     #^RXPT,R2               ;; SET XPT'S NAME IN RAD50
001074                          CALL    @PDVID                  ;; GET IT'S PDV INDEX           ; RJK10
001100  103423                  BCS     10$                     ;; IF CS, XPT NOT IN SYSTEM
001102  120264  000055          CMPB    R2,L$PDV(R4)            ;; OWNED BY XPT ?
001106  001020                  BNE     10$                     ;; IF NE, NO
001110  067702  000000G         ADD     @PDVTA,R2               ;; POINT INTO PDV TABLE         ; RJK10
001114  011202                  MOV     (R2),R2                 ;; GET PDV ADDRESS
001116  016205  000016          MOV     Z.DAT(R2),R5            ;; POINT TO XPT'S DATA BASE
001122  005765  000020          TST     N$PLD+2(R5)             ;; HAS THE PLB BEEN ASSIGNED YET ;09
001126  001410                  BEQ     10$                     ;; IF EQ, NO.                    ;09
001130  116401  000054          MOVB    L$CHN(R4),R1            ;; GET THE CHANNEL NUMBER ASSIGNED
001134  006301                  ASL     R1                      ;; MAKE IT A WORD INDEX
001136  066501  000020          ADD     N$PLD+2(R5),R1          ;; POINT TO ADDRESS OF PLB
001142  011166  000000G         MOV     (R1),R$R2(SP)           ;; RETURN PLB ADDRESS IN SAVED R2
001146  000405                  BR      20$                     ;; RETURN

001150                  10$:    RETC    R0                      ;; SET USER C-BIT
001162                  20$:    RETURN

                                .IF NDF S$$BAS & R$$STR & R$$PRO
```

```
                                          .SBTTL  SETNAM - Set an image name field
                                  ;+
                                  ; **-SETNAM-SET AN IMAGE NAME FIELD
                                  ;
                                  ; THIS ROUTINE BUILDS AN IMAGE NAME FIELD, DELETING TRAILING
                                  ; SPACES.
                                  ;
                                  ; INPUTS:
                                  ;       R0 = POINTER TO 6 CHAR SPACE-FILLED NAME
                                  ;       R3 = ADDRESS OF NEXT BYTE IN OUTPUT BUFFER
                                  ;
                                  ; OUTPUTS:
                                  ;       R3 = ADDRESS OF NEXT BYTE IN OUTPUT BUFFER
                                  ;
                                  ; REGISTERS:
                                  ;       R4, R5 ARE PRESERVED
                                  ;-

003320                            SETNAM:
003320  010302                            MOV     R3,R2                   ; COPY BUFFER POINTER
003322  112723  000007                    MOVB    #7,(R3)+                ; ASSUME NAME IS 6 CHARS
003326  112023                            MOVB    (R0)+,(R3)+             ; SET NAME IN BUFFER
003330  112023                            MOVB    (R0)+,(R3)+             ; ...
003332  112023                            MOVB    (R0)+,(R3)+             ; ...
003334  112023                            MOVB    (R0)+,(R3)+             ; ...
003336  112023                            MOVB    (R0)+,(R3)+             ; ...
003340  112023                            MOVB    (R0)+,(R3)+             ; ...
003342  105312                    10$:    DECB    (R2)                    ; DEC FIELD COUNT
003344  124327  000040                    CMPB    -(R3),#'                ; IS THE LAST CHARACTER A SPACE ?
003350  001774                            BEQ     10$                     ; IF EQ, YES - DELETE IT
003352  005203                            INC     R3                      ; ELSE, SKIP OVER IT
003354                                    RETURN

        000001                            .END
```

```
                                                                      B 14
CIRECH      CREATED BY  MACRO  ON 29-JUN-85 AT 12:16       PAGE 7
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE              REFERENCES

S.RDE     000034           #6-74
S.RTY     000001           #6-74
S.STA     000000           #6-74
S.STPN    000034            11-388
S.TLZ     000012           #6-74
S.TMO     000040           #6-74
S.XDE     000036           #6-74
S.XID     000002           #6-74
S.XMB     000030           #6-74
S.XMP     000024           #6-74
TF.FFE  = 000004           #6-74
TF.ROF  = 000001           #6-74
TF.RTO  = 000002           #6-74
TF.TAE  = 000010           #6-74
TF.TOF  = 000004           #6-74
TF.TTO  = 000010           #6-74
TF.URE  = 000001           #6-74
TF.USA  = 000002           #6-74
TR.ALF    000102           #6-74
TR.DEV    000100           #6-74
TR.LEN  = 000210           #6-74
TR.MST    000074           #6-74
TR.PRE    000076           #6-74
TR.TLZ    000072           #6-74
TX.NXM  = 000002           #6-74
TX.OVR  = 000001           #6-74
TX.TMO  = 000004           #6-74
T$FLAG    000044           #6-63        13-480      17-773      17-775
T$LIF     000013           #6-63
T$LIFL    000013           #6-63
T$LIFO    000013           #6-63
T$LIFS    000013           #6-63
T$LIN     000000           #6-63        17-768
T$LIPS    000006           #6-63
T$LLD     000012           #6-63
T$LLDC    000045           #6-63
T$LLDL    000012           #6-63
T$LLDO    000012           #6-63
T$LLDS    000012           #6-63
T$LLEN    000046           #6-63        17-770
T$LOPR    000002           #6-63
T$LTCL    000024           #6-63
T$LTIM    000026           #6-63
T$LTPR    000014           #6-63
T$LTPS    000020           #6-63
T$NAPL    000004           #6-63
T$NFE     000000           #6-63
T$NLEN    000010           #6-63
T$NNUL    000002           #6-63
T$NOPL    000006           #6-63
T$NRNI    000042           #6-63
T$NRPL    000005           #6-63
```

```
                                                                      C 14
CIRECH      CREATED BY  MACRO  ON 29-JUN-85 AT 12:16       PAGE 8
```

```
    115                                                   .SBTTL  CIRECO - READ/ZERO LINE COUNTERS
    116                                         ;+
    117                                         ; **-CIRECO-READ/ZERO LINE COUNTERS
    118                                         ;
    119                                         ; THIS ROUTINE DISPATCHES TO A ROUTINE TO READ OR ZERO A LINE'S
    120                                         ; COUNTERS BASED ON THE LINE'S DLC.
    121                                         ;
    122                                         ; INPUTS:
    123                                         ;         R3 = ADDRESS OF NEXT FREE BYTE IN OUTPUT BUFFER
    124                                         ;         R4 = ADDRESS OF CONTEXT BLOCK
    125                                         ;
    126                                         ; OUTPUTS:
    127                                         ;         R3 = ADDRESS OF NEXT FREE BYTE IN OUTPUT BUFFER
    128                                         ;
    129                                         ; REGISTERS:
    130                                         ;         R4 IS PRESERVED
    131                                         ;-
    132
    133 000002                                  CIRECO::
    134                                         ;
    135                                         ; LOOK FOR LINE'S DLC IN THE DLC TABLE AND DISPATCH TO THE
    136                                         ; COUNTER ROUTINE FOR THE DLC.
    137                                         ;
    138 000002  016400  000042                          MOV     L$SLT(R4),R0            ; GET SLT ADDRESS
    139
    140                                                 .IF NDF R$$11S & R$$PRO
    141
    142 000006  022764  177777  000050                  CMP     #-1,L$TPT(R4)           ; IS IT A PVC CIRCUIT?
    143 000014  001003                                  BNE     5$                      ; BR IF NO
    144 000016                                          CALL    PVCCOU                  ; READ PVC COUNTERS
    145 000022  000444                                  BR      30$                     ; AND EXIT
    146
    147                                                 .ENDC   ; NDF R$$11S & R$$PRO
    148
    149 000024  032760  040000  000000  5$:             BIT     #LF.RDY,L.FLG(R0)       ; IS THE LINE LOADED ?
    150 000032  001440                                  BEQ     30$                     ; IF EQ, NO - SKIP COUNTERS
    151 000034  116002  000003                          MOVB    L.DLC(R0),R2            ; GET THE DLC PDV INDEX
    152 000040  067702  000000G                         ADD     @PDVTA,R2               ; POINT TO THE PDV ADDRESS      ; RJK02
    153 000044  011202                                  MOV     (R2),R2                 ; GET THE PDV ADDRESS           ;**-1
    154 000046  032762  001000  000010                  BIT     #ZF.COU,Z.FLG(R2)       ; DOES THE DLC SUPPORT COUNTERS ?
    155 000054  001427                                  BEQ     30$                     ; IF EQ, NO - SKIP LOOKING
    156 000056  012705  000000'                         MOV     #DLCTBL,R5              ; POINT TO START OF DLC TABLE
    157 000062  026225  000004          10$:            CMP     Z.NAM(R2),(R5)+         ; IS THIS THE DLC FOR THE LINE ?
    158 000066  001002                                  BNE     20$                     ; IF NE, NO - KEEP LOOKING
    159 000070                                          CALL    @(R5)+                  ; ELSE, CALL DLC LINE COUNTER ROUTINE
    160 000072  000420                                  BR      30$                     ; AND RETURN
    161 000074  005725                  20$:            TST     (R5)+                   ; SKIP OVER ADDRESS
    162 000076  005715                                  TST     (R5)                    ; END OF TABLE ?
    163 000100  001370                                  BNE     10$                     ; IF NE, NO
    164 000102  032762  020000  000010                  BIT     #ZF.MAN,Z.FLG(R2)       ; SUPPORT FOR PROCESS LEVEL COUNTERS ?
    165 000110  001402                                  BEQ     25$                     ; IF EQ, NO
    166 000112                                          CALLR   $PRCCO                  ; ELSE GET COUNTERS FROM PROCESS
    167 000116  016403  000060          25$:            MOV     L$BUF(R4),R3            ; RESTORE BUFFER POINTER
    168 000122  112723  177770                          MOVB    #ME.UCO,(R3)+           ; INDICATE UNRECOGNIZED COMPONENT
    169 000126  112723  000003                          MOVB    #ME$CIR,(R3)+           ; STORE ERROR DETAIL
    170 000132  105023                                  CLRB    (R3)+                   ; STORE HIGH BYTE
    171
```

```
        .TITLE  CIREIN - CIRCUIT READ INFORMATION INITIALIZE
        .IDENT  /V05.00/
        .ENABL  LC
        .NLIST  CND
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CIRCUIT READ INFORMATION INITIALIZE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  30-JUN-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
MF$TES  000022      MP$ALB  000036      MP$DLI= 001466      MP$MAR  001635      MP$PCO  000024
MF$TRA  000301      MP$AMC  001640      MP$DLT  002200      MP$MAV  006321      MP$PCT  000144
MF$TRI  000021      MP$AMH  001641      MP$DRO  001441      MP$MBN  001636      MP$PDT  002114
MF$ZER  000025      MP$ANB  004420      MP$DST  000454      MP$MBR  001637      MP$PHA  000170
ML$ALL  100000      MP$ASB  000024      MP$DTE  002140      MP$MBU  001642      MP$PHY  000012
ML$CLS  000000      MP$ASC  004432      MP$DTY= 001452      MP$MCB  000156      MP$PLI  000025
ML$CON  000001      MP$AUS  000512      MP$DUA  000207      MP$MCO  001632      MP$PLN  002140
ML$EXT  000001      MP$BBT  002165      MP$DUC  000210      MP$MDE  004421      MP$PLO  000012
ML$FIL  000002      MP$BDF  002164      MP$DUM  000202      MP$MHO  001633      MP$PMC  004374
ML$FIR  000000      MP$BFQ  002121      MP$DUP  002127      MP$MLB  000202      MP$PNT  002126
ML$INT  000000      MP$BLK  001616      MP$DVC  002114      MP$MLK  001306      MP$PRI  004411
ML$KNO  140000      MP$BLO  001452      MP$DWE  001321      MP$MLN  001631      MP$PRO  002130
ML$MON  000003      MP$BMX  002176      MP$ELT  000157      MP$MLP  006333      MP$PSS  001763
ML$SYS  000002      MP$BNP  006330      MP$ETY  001605      MP$MRB  002171      MP$PST  001762
ML$TOP  000001      MP$BRT  001620      MP$EVE  000311      MP$MRO  001605      MP$RET  001631
MN$UNL  000377      MP$BSA  002176      MP$FNC  001752      MP$MRP  006322      MP$RFA  001323
MO$ACC  000200      MP$BSD  002203      MP$GDT  002222      MP$MRT  002153      MP$RMX  002201
MO$ADD  000002      MP$BSI  002200      MP$GNM  002223      MP$MRV  006320      MP$RPA  004374
MO$ALA  000004      MP$BSP  006331      MP$GRO  002115      MP$MSB  000170      MP$RPR  001606
MO$ALI  000000      MP$BUF  000170      MP$GRP  000541      MP$MVE  000145      MP$RRT  002213
MO$CHA  000040      MP$BUP  006324      MP$GTY  002224      MP$MVI  001634      MP$RST  002212
MO$CIR  000003      MP$BUS  001643      MP$HAD  001757      MP$MVR  001751      MP$RSV  001754
MO$CLE  000100      MP$CAC  001750      MP$HBT  002142      MP$MWN  002154      MP$RSZ  001756
MO$COU  000060      MP$CAS  001762      MP$HDD  000162      MP$MXB  002172      MP$RTI  001616
MO$DAC  000000      MP$CAT  002210      MP$HTM  001612      MP$MXC  000466      MP$RTM  001611
MO$DEF  000000      MP$CCS  004406      MP$HWA  002210      MP$MXR  001630      MP$RVE  001604
MO$DPR  000000      MP$CHN  002141      MP$IAT  001322      MP$MXW  002143      MP$RVT  000156
MO$ENT  000017      MP$CIR  000144      MP$IDE  000144      MP$NAA  006335      MP$SAC  000514
MO$EVE  000100      MP$CLK  002131      MP$IDP  006327      MP$NAC  001130      MP$SAD  000543
MO$INF  000160      MP$CLN  002126      MP$IHO  000215      MP$NAP  006326      MP$SCA  000310
MO$INS  000002      MP$CLT  002211      MP$INA  002177      MP$NCT  000240      MP$SCO  000144
MO$LIN  000001      MP$CMB  002142      MP$IND  002204      MP$NET  002114      MP$SCT  002176
MO$LOG  000001      MP$CMK  000537      MP$INI  002201      MP$NLI  000765      MP$SDU  000203
MO$MJR  000031      MP$CMX  002153      MP$ITI  000776      MP$NNA  000764      MP$SDV  000160
MO$NAM  000001      MP$CND  000310      MP$LAA  000231      MP$NOD  000500      MP$SEH  004401
MO$NIC  000023      MP$CNU  001753      MP$LAN  000234      MP$NRA  006334      MP$SER  000144
MO$NOD  000000      MP$COB  000311      MP$LAR  000202      MP$NRB  000214      MP$SET  000000
MO$OFF  000001      MP$CON  002126      MP$LAS  000012      MP$NSA  001617      MP$SGZ  001644
MO$ON   000000      MP$COS  001604      MP$LBS  000203      MP$NTI  002141      MP$SID  000176
MO$OPT  000001      MP$CPF  000230      MP$LCO  000226      MP$NUM  001642      MP$SIN  000310
MO$PER  000200      MP$CPL  000232      MP$LCT  000156      MP$NVE  001274      MP$SLI  000156
MO$PRO  000002      MP$CPT  000226      MP$LHL  000232      MP$NXN= 001476      MP$SLO  000171
MO$PRS  000001      MP$CPU  000161      MP$LLE  000227      MP$OAC  000632      MP$SMX  002202
MO$REA  000200      MP$CSZ  001755      MP$LLO  004432      MP$OAN  000620      MP$SND  000500
MO$SET  000000      MP$CUS  002127      MP$LMB  002152      MP$OCO  000776      MP$SNP  006332
MO$STA  000020      MP$CVA  000540      MP$LMX  002200      MP$OHO  000214      MP$SNU  000542
MO$SUM  000000      MP$DAL  002570      MP$LNA  000144      MP$ONA  000764      MP$SOB  000524
MO$TSK  000000      MP$DCO= 001464      MP$LNO  000233      MP$ONR  002114      MP$SPA  000157
MO$VOL  000000      MP$DDT  002177      MP$LOA  000170      MP$OQL  001453      MP$SPR  000536
MO$ZER  000000      MP$DEL  001131      MP$LUG  004514      MP$OTI  000777      MP$SPS  000513
MP$ACB  000012      MP$DES  000156      MP$LOO  000620      MP$OTY  001022      MP$STA  000000
MP$ACC  000514      MP$DEV  002114      MP$LTM  001613      MP$OUS  000777      MP$STI  002140
MP$ACT  004526      MP$DFA  001320      MP$LTY  004533      MP$OVE  001010      MP$STT  002201
MP$ADD  000766      MP$DHO= 001465      MP$LWI  000230      MP$OWN  004374      MP$STY  000175
MP$ADJ  001440      MP$DIA  000173      MP$MAC  004420      MP$PAR  000036      MP$SUB  000001
MP$ADP  006325      MP$DLB  002571      MP$MAD  001630      MP$PAS  000513      MP$SUR  000156
MP$ADS  004406      MP$DLG  004521      MP$MAP  006323      MP$FCH  002152      MP$SUS  000512
```

```
MACRO CROSS REFERENCE                                CREF   04.00

MACRO NAME      REFERENCES

ASL$            #5-57      27-1046
CALL             10-546    10-565    13-653    13-662    13-676    13-688    13-689    22-925    22-932    23-951
                 23-958    28-1070   29-1088   30-1106   31-1124   32-1142   33-1160   34-1178   35-1196   36-1214
                 37-1232   38-1250   39-1268   40-1286   41-1304
CICCX$          #5-57      6-72
DBGTP$          #8-82      8-88     #8-90     #8-92      8-94     #8-95      8-97     #8-98      8-100    #8-101
                 8-103    #8-104     8-106    #8-107     8-109    #8-110     8-112    #8-113     8-115    #8-116
                 8-118    #8-119     8-121    #8-122     8-124    #8-125     8-127    #8-128     8-130    #8-131
                 8-133    #8-134     8-136    #8-137     8-139    #8-140     8-142    #8-143     8-145    #8-146
                 8-148    #8-149     8-151    #8-152     8-154    #8-155     8-157    #8-158     8-160    #8-161
                 8-163    #8-164     8-166    #8-167     8-169    #8-170     8-172    #8-173     8-175    #8-176
                 8-178    #8-179     8-181    #8-182     8-184    #8-185     8-187    #8-188     8-189     8-194
                #8-195     8-197    #8-198     8-200    #8-201     8-207    #8-208     9-216     9-219     9-226
                 9-228    #9-229    #9-231    #9-233     9-239     9-241     9-246     9-248    #9-249     9-254
                 9-256    #9-257     9-262     9-264    #9-265     9-270     9-272     9-277     9-279    #9-280
                 9-282    #9-283     9-285    #9-286     9-288     9-293     9-295     9-300     9-302     9-307
                 9-309    #9-310    #9-311     9-316     9-318     9-323     9-325     9-330     9-332     9-337
                 9-339     9-344     9-346     9-351     9-353     9-358     9-360     9-365     9-367     9-372
                 9-374     9-379     9-381     9-386     9-388     9-393     9-395     9-400     9-402     9-407
                 9-409     9-414     9-416    #9-417    #9-418     9-423     9-425     9-430     9-432     9-437
                 9-439    #9-440    #9-441    #9-442     9-447     9-449    #9-450     9-455     9-457     9-462
                 9-464     9-469     9-471    #9-472     9-477     9-479     9-484     9-487     9-492     9-497
EXT$$$          #8-88     #8-90     #8-94     #8-97     #8-100    #8-103    #8-106    #8-109    #8-112    #8-115
                #8-118    #8-121    #8-124    #8-127    #8-130    #8-133    #8-136    #8-139    #8-142    #8-145
                #8-148    #8-151    #8-154    #8-157    #8-160    #8-163    #8-166    #8-169    #8-172    #8-175
                #8-178    #8-181    #8-184    #8-187    #8-194    #8-197    #8-200    #8-208    #9-219    #9-226
                #9-228    #9-229    #9-231    #9-239    #9-246    #9-248    #9-249    #9-254    #9-256    #9-262
                #9-264    #9-270    #9-277    #9-279    #9-280    #9-282    #9-285    #9-288    #9-293    #9-300
                #9-307    #9-309    #9-310    #9-316    #9-318    #9-323    #9-325    #9-330    #9-332    #9-337
                #9-339    #9-344    #9-346    #9-351    #9-353    #9-358    #9-360    #9-365    #9-367    #9-372
                #9-374    #9-379    #9-381    #9-386    #9-388    #9-393    #9-395    #9-400    #9-402    #9-407
                #9-409    #9-414    #9-416    #9-417    #9-423    #9-425    #9-430    #9-432    #9-437    #9-439
                #9-440    #9-441    #9-447    #9-449    #9-455    #9-457    #9-462    #9-464    #9-469    #9-471
                #9-477    #9-479    #9-487    #9-492    #9-497
FIELD$          #8-82      8-86      8-93      8-96      8-99      8-102     8-105     8-108     8-111     8-114
                 8-117     8-120     8-123     8-126     8-129     8-132     8-135     8-138     8-141     8-144
                 8-147     8-150     8-153     8-156     8-159     8-162     8-165     8-168     8-171     8-174
                 8-177     8-180     8-183     8-186     8-193     8-196     8-199     8-206     9-215     9-218
                 9-225     9-227     9-238     9-240     9-245     9-247     9-253     9-255     9-261     9-263
                 9-269     9-271     9-276     9-278     9-281     9-284     9-287     9-292     9-294     9-299
                 9-301     9-306     9-308     9-315     9-317     9-322     9-324     9-329     9-331     9-336
                 9-338     9-343     9-345     9-350     9-352     9-357     9-359     9-364     9-366     9-371
                 9-373     9-378     9-380     9-385     9-387     9-392     9-394     9-399     9-401     9-406
                 9-408     9-413     9-415     9-422     9-424     9-429     9-431     9-436     9-438     9-446
                 9-448     9-454     9-456     9-461     9-463     9-468     9-470     9-476     9-478     9-483
                 9-486     9-491     9-496     9-501
HDRDF$          #5-59      5-61
IMSG$           #5-57      8-82
ITRAN$          #8-82      8-88      8-90      8-94      8-97      8-100     8-103     8-106     8-109     8-112
                 8-115     8-118     8-121     8-124     8-127     8-130     8-133     8-136     8-139     8-142
                 8-145     8-148     8-151     8-154     8-157     8-160     8-163     8-166     8-169     8-172
                 8-175     8-178     8-181     8-184     8-187     8-189     8-194     8-197     8-200     8-207
```

```
000242  103424                            BCS     20$                     ; BR if not valid operation
                                  ;
                                  ; DISPATCH TO SET OR CLEAR ROUTINES
                                  ;
000244  005764  000070                    TST     L$FL2(R4)               ; ANY WORK TO DO?
                                          .IF NDF R$$11S
000250  001404                            BEQ     15$                     ; BR IF NO
                                          .IFF    ; NDF R$$11S
                                          BEQ     20$                     ; BR IF NO
                                          .IFTF   ; NDF R$$11S
000252  012705  000000'                   MOV     #DISTBL,R5              ; POINT TO START OF TABLE
000256                                    CALL    DSPTCH                  ; DISPATCH TO PROPER ROUTINE
000262                            15$:
                                          .IFT    ; NDF R$$11S
000262  016464  000066  000070            MOV     L$FL1(R4),L$FL2(R4)     ; Get second flags word
000270  001404                            BEQ     18$                     ; BR if no work to do
000272  012705  000056'                   MOV     #DISTB1,R5              ; Point to start of table
000276                                    CALL    DSPTCH                  ; Dispatch to proper routine
000302                            18$:

                                          .IF NDF R$$PRO  ; PRO/DECNET

000302  005764  000076                    TST     L$FLX(R4)               ; Check X.25 circuit flags
000306  001402                            BEQ     20$                     ; Branch if no X.25 parameters
000310                                    CALL    CICHPX                  ; Change other X.25 circuit parameters

                                          .ENDC   ; NDF R$$PRO

                                          .ENDC   ; NDF R$$11S
                                  ;
                                  ; USE THE STATUS IN R0 TO BUILD RETURN MESSAGE AND C-BIT RETURN
                                  ;
000314  020027  177600            20$:    CMP     R0,#ME.DON              ; ARE WE DONE ?
000320  001006                            BNE     25$                     ; IF NE, NO
000322  005064  000064                    CLR     L$FLG(R4)               ; CLEAR INIT FLAG IN FLAGS WORD
                                          .IF NDF R$$11S
000326  005064  000066                    CLR     L$FL1(R4)               ; ...

                                          .IF NDF R$$PRO  ; PRO/DECNET

000332  005064  000076                    CLR     L$FLX(R4)               ; ... (X.25)

                                          .ENDC   ; NDF R$$PRO

                                          .ENDC   ; NDF R$$11S
000336  016403  000060            25$:    MOV     L$BUF(R4),R3            ; GET BUFFER POINTER
000342  110023                            MOVB    R0,(R3)+                ; SET THE RETURN CODE
000344  100402                            BMI     27$                     ; IF MI, L$PAR CONTAINS DETAIL
000346  005064  000072                    CLR     L$PAR(R4)               ; ELSE, NO DETAIL TO RETURN
000352  116423  000072            27$:    MOVB    L$PAR(R4),(R3)+         ; SET THE ERROR DETAIL BYTES
000356  116423  000073                    MOVB    L$PAR+1(R4),(R3)+       ; ...
000362  105023                            CLRB    (R3)+                   ; ASSUME NO ERROR MESSAGE TEXT

                                          .IF NDF R$$PRO  ; PRO/DECNET

000364  016405  000056                    MOV     L$MSG(R4),R5            ; GET MESSAGE POINTER
000370  001405                            BEQ     40$                     ; IF EQ, NONE
```

```
                                                                 C  4
CICHOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:13      PAGE 2
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE                REFERENCES

LN.CLO   = 000000             #6-68
LN.DUM   = 000005             #6-68
LN.LOA   = 000004             #6-68
LN.LOO   = 000003             #6-68
LN.OAU   = 000003             #6-68
LN.OFF   = 000001             #6-68
LN.ON    = 000000             #6-68
LN.OOP   = 000004             #6-68
LN.OPE   = 000001             #6-68
LN.REF   = 000002             #6-68
LN.SER   = 000002             #6-68
LN.STA   = 000017             #6-68
LN.SUB   = 000360             #6-68
LN.TRI   = 000006             #6-68
LO.CON     000006             #6-66
LO.CSR     000006             #6-66
LO.CTL     000014             #6-66
LO.LIN     000004             #6-66
LO.PAR     000010             #6-66
LO.PRI     000010             #6-66
LO.SCR     000014             #6-66
LO.STA     000015             #6-66
LO.VCT     000012             #6-66
LP$MPT   = 000010             #7-73
LP$MUX   = 000004             #7-73
LP$NXC   = 100000             #7-73
LP$TRB   = 000002             #7-73
LP$UNT   = 000001             #7-73
LP$WCN   = 000040             #7-73        7-73
LP$WDV   = 000020             #7-73        7-73
LP$WLD   = 000360             #7-73        9-252
LP$WTR   = 000200             #7-73        7-73
LP$WUN   = 000100             #7-73        7-73
LR.AST     000026             #6-66
LR.CIR     000002             #6-66
LR.CTL     000004             #6-66
LR.EFN     000004             #6-66
LR.LIN     000002             #6-66
LR.PRM     000030             #6-66
LR.PRO     000002             #6-66
LR.STS     000000             #6-66
LR.TCB     000002             #6-66
LR.UNT     000005             #6-66
LS$ACT   = 000040             #7-73        8-92
LS$BBT   = 010000             #7-73        8-99
LS$BLK   = 001000             #7-73        7-73
LS$BSA   = 000001             #7-73        8-112
LS$BSD   = 000002             #7-73        8-113
LS$BSI   = 000004             #7-73        8-114
LS$CHN   = 000001             #7-73        7-73
LS$CMB   = 000002             #7-73        7-73         7-73         7-73
LS$COS   = 000004             #7-73        8-88


                                                                 D  4
CICHOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:13      PAGE 3
```

```
        000125                          L$INA:  .BLKB   1               ; DMP ACTIVE INCREMENT
        000126                          L$IND:  .BLKB   1               ; DMP DYING INCREMENT
        000127                          L$INI:  .BLKB   1               ; DMP INACTIVE INCREMENT
        000130                          L$TH1:  .BLKB   1               ; DMP DEAD THRESHOLD
        000131                          L$TH2:  .BLKB   1               ; DMP DYING THRESHOLD
        000132                          L$TH3:  .BLKB   1               ; DMP INACTIVE THRESHOLD
        000133                          L$MXB:  .BLKB   1               ; DMP MAXIMUM BLOCKS
        000134                          L$NTL:  .BLKW   13.             ; NTL MESSAGE BUFFER
        000166                          L$SCR:  .BLKW   15.             ; SCRATCH BUFFER
        000224                          L$LCT:  .BLKW   1               ; COUNTER TIMER
        000226                          L$HTM:  .BLKW   1               ; HELLO TIMER
        000230                          L$SER:                          ; SERVICE
        000230                          L$LTM:  .BLKW   1               ; LISTEN TIMER
        000232                          L$XCN:  .BLKW   1               ; X25 Logical Channel Number (LCN)
        000234                          L$LMB:                          ; X25 Max Block (Line)
        000234                          L$CMB:  .BLKW   1               ; X25 Max Data (Circuit)
        000236                          L$NUML:                         ; DLM Number length
        000236                          L$DTEL: .BLKW   1               ; X25 DTE length (Circuit)
        000240                          L$NUM:                          ; DLM Number
        000240                          L$DTE:  .BLKB   8.              ; X25 DTE (Circuit)
        000250                          L$DTEP: .BLKW   1               ; X25 Pointer to DTE descriptor (Circuit)
        000252                          L$MWN:                          ; X25 Max Window (Line)
        000252                          L$MXW:  .BLKB   1               ; X25 Max Window (Circuit)
        000253                          L$MRT:                          ; X25 Max Retransmits (Line)
        000253                          L$MXR:  .BLKB   1               ; DLM Max Recalls (Circuit)
        000254                          L$HBT:  .BLKW   1               ; X25 Holdback Timer (Line)
        000256                          L$NTI:                          ; X25 Retransmit Timer (Line)
        000256                          L$RET:  .BLKW   1               ; DLM Recall Timer (Circuit)
        000260                          L$CUS:  .BLKW   1               ; DLM Usage
        000262                          L$BLK:  .BLKW   1               ; DLM blocking state
                                                .IF DF  R$$PRO  ;PRO/DECnet
                                        L$LTY:  .BLKW   1               ; Loopback Type
                                        L$MDT:  .BLKW   1               ; Modem Test
                                                .ENDC   ; DF R$$PRO
                                                .EVEN
        000264                          L$LEN:                          ; LENGTH OF CONTEXT AREA
        000000                                  .PSECT
                                        ;
                                        ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                        ;
                000001                          LF$REA = 1              ; READ COUNTERS OPERATION
                000002                          LF$ZER = 2              ; ZERO COUNTERS OPERATION
                000004                          LF$SKP = 4              ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
                000010                          LF$VR2 = 10             ; CONNECTED TO VERSION 2.0 NCP
                040000                          LF$MLT  = 40000         ; MULTIPLE ADJACENCY FLAG
                100000                          LF$SEG  = 100000        ; SEGMENTED RESPONSE IN PROGRESS
                                        ;
                                        ; PARSE FLAG DEFINITIONS (L$PFG)
                                        ;
                000001                          LP$UNT = 1              ; UNIT NUMBER FOUND
                000002                          LP$TRB = 2              ; TRIBUTARY NUMBER FOUND
                000004                          LP$MUX = 4              ; DEVICE IS MUX
                000010                          LP$MPT = 10             ; LINE IS MULTIPOINT
```

```
    487                                     ;
    488 001466  126427  000100  000000  140$:   CMPB    L$STA(R4),#MS$ON        ; IS IT "ON" ?
    489 001474  001003                          BNE     150$                    ; IF NE, NO
    490 001476                                  CALL    STAON                   ; ELSE, SET STATE ON
    491 001502  000413                          BR      180$                    ; AND RETURN
    492 001504  126427  000100  000002  150$:   CMPB    L$STA(R4),#MS$SER       ; IS IT SERVICE ?
    493 001512  001003                          BNE     160$                    ; IF NE, NO
    494 001514                                  CALL    STASER                  ; ELSE , SET STATE TO SERVICE
    495 001520  000404                          BR      180$                    ; AND RETURN
    496 001522                          160$:   CALL    STAOFF                  ; ELSE, MUST BE OFF
    497 001526  000401                          BR      180$                    ;
    498                                         .ENDC
    499
    500 001530  000261                  170$:   SEC                             ; RETURN WITH ERROR
    501 001532                          180$:   RETURN
    502
    503                                         .IF NDF R$$11S
```

```
                                                  .SBTTL  SETOWN - SET CIRCUIT OWNER
                                         ;+
                                         ; **-SETOWN-SET CIRCUIT OWNER
                                         ;
                                         ; THIS ROUTINE IS CALLED TO SET THE OWNER FOR A CIRCUIT.
                                         ;
                                         ; INPUTS:
                                         ;        R4 = ADDRESS OF CONTEXT AREA WITH:
                                         ;             L$OWN(R4) = NEW OWNER'S PDV INDEX (IF ENTRY AT SETOWN)
                                         ;
                                         ; OUTPUTS:
                                         ;        THE OWNER IS SET
                                         ;
                                         ; REGISTERS:
                                         ;        R4, R5 ARE PRESERVED
                                         ;-

                                                  .IF NDF R$$11S

 000000                                           .PSECT  $$$UNM,RO,I             ; FORCE TO APR 6

                                                  .ENDC   ; NDF R$$11S

 000000                                  SETOWN::
 000000  012700  177752                           MOV     #ME.PNA,R0              ; ASSUME PVC
 000004  022764  177777  000050                   CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
 000012  001502                                   BEQ     70$                     ; BR IF YES - ERROR
 000014  012700  177767                           MOV     #ME.IID,R0              ; ASSUME INVALID CIRCUIT-ID
 000020  012764  000003  000072                   MOV     #ME$CIR,L$PAR(R4)       ; ...
 000026  132764  000010  000041                   BITB    #LP$MPT,L$PFG(R4)       ; IS THIS A MULTI-POINT CIRCUIT ?
 000034  001404                                   BEQ     10$                     ; IF EQ, NO
 000036  132764  000002  000041                   BITB    #LP$TRB,L$PFG(R4)       ; ELSE, WAS A TRIB NUMBER GIVEN ?
 000044  001465                                   BEQ     70$                     ; IF EQ, NO - ERROR
                                         ;
                                         ; TRY TO ASSIGN LUN TO NX:
                                         ;
 000046  012700  177777                  10$:     MOV     #ME.FUN,R0              ; ASSUME ILLEGAL FUNCTION
 000052                                           CALL    DLXLUN                  ; TRY TO ASSIGN LUN
 000056  103457                                   BCS     60$                     ; IF CS, NO DLX - ERROR
                                         ;
                                         ; MAKE SURE THE CIRCUIT IS LOADED, BUT OFF
                                         ;
 000060  016402  000042                           MOV     L$SLT(R4),R2            ; GET THE SLT ADDRESS
 000064  016403  000050                           MOV     L$TPT(R4),R3            ; POINT TO TRIB FLAGS
 000070  012700  177765                           MOV     #ME.CST,R0              ; ASSUME CIRCUIT IS IN WRONG STATE
 000074  032762  040000  000000                   BIT     #LF.RDY,L.FLG(R2)       ; IS THE CIRCUIT LOADED ?
 000102  001445                                   BEQ     60$                     ; IF EQ, NO - IN WRONG STATE
 000104  105762  000014                           TSTB    L.NSTA(R2)              ; MULTIPOINT CIRCUIT ?
 000110  001005                                   BNE     20$                     ; IF NE, YES
 000112  032762  100000  000000                   BIT     #LF.ACT,L.FLG(R2)       ; ELSE, IS THE CIRCUIT ON ?
 000120  001036                                   BNE     60$                     ; IF NE YES - IN WRONG STATE
 000122  000404                                   BR      30$                     ; ELSE, CONTINUE
 000124  132763  000200  000000  20$:     BITB    #SF.ACT,S.FLG(R3)       ; ELSE, IS THE STATION ON ?
 000132  001031                                   BNE     60$                     ; IF NE, YES - CIRCUIT IS IN WRONG STATE
                                         ;
                                         ; ISSUE REASSIGN REQUEST TO DLX
                                         ;
```

```
                                                                    C  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

LF$REA  = 000001           #6-86
LF$SEG  = 100000           #6-86
LF$SKP  = 000004           #6-86
LF$VR2  = 000010           #6-86
LF$ZER  = 000002           #6-86
LF.ACT  = 100000           #5-73     13-358    13-439    18-766    19-825    20-877    21-980
LF.BRO  = 000400           #5-73     13-366    13-441    15-596
LF.BWT  = 000007           #5-73
LF.ENA  = 002000           #5-73
LF.LPB  = 001000           #5-73
LF.MDC  = 000100           #5-73
LF.MFL  = 004000           #5-73
LF.MTP  = 000020           #5-73     13-441    19-810
LF.PAC  = 000200           #5-73
LF.RDY  = 040000           #5-73     9-172     12-291    13-348    20-869    21-976
LF.REA  = 010000           #5-73
LF.SER  = 000040           #5-73     24-1130   24-1133
LF.TIM  = 000010           #5-73
LF.UNL  = 020000           #5-73
LF.X2P  = 000000           #5-73
LINOFF    002414 R          17-705  #19-789
LINON     002174 R          17-698  #18-722
LLCRS   = ******   GX       18-749   19-800
LN.CLO  = 000000           #5-73
LN.DUM  = 000005           #5-73
LN.LOA  = 000004           #5-73
LN.LOO  = 000003           #5-73
LN.OAU  = 000003           #5-73
LN.OFF  = 000001           #5-73     18-735
LN.ON   = 000000           #5-73
LN.OOP  = 000004           #5-73
LN.OPE  = 000001           #5-73
LN.REF  = 000002           #5-73     14-538    14-540    15-616
LN.SER  = 000002           #5-73
LN.STA  = 000017           #5-73
LN.SUB  = 000360           #5-73
LN.TRI  = 000006           #5-73
LO.CON    000006           #5-71
LO.CSR    000006           #5-71
LO.CTL    000014           #5-71
LO.LIN    000004           #5-71
LO.PAR    000010           #5-71
LO.PRI    000010           #5-71
LO.SCR    000014           #5-71
LO.STA    000015           #5-71
LO.VCT    000012           #5-71
LP$MPT  = 000010           #6-86     10-199    20-863    21-960    23-1093
LP$MUX  = 000004           #6-86
LP$NXC  = 100000           #6-86
LP$TRB  = 000002           #6-86     13-368    20-865    21-962    23-1084   23-1089
LP$UNT  = 000001           #6-86
LP$WCN  = 000040           #6-86     6-86
```

```
                                                                    D  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14     PAGE 3
```

```
                                        .SBTTL  Context area definition

        000000                          CICCX$  LIST
                                ;
                                ; CONTEXT AREA OFFSET DEFINITIONS
                                ;
        000004                          .ASECT
                  000000                . = 0
                  000020                L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                ;
                                ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                ; as in the CIRCX$ and LOCCX$ macros.
                                ;
        000000                  L$NAM:
        000000                  L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                  L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                  L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                  L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                  000020                . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                  L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                  L$TRB:  .BLKB   1               ; TRIBUTARY NUMBER
        000041                  L$PFG:  .BLKB   1               ; PARSE FLAGS
        000042                  L$SLT:  .BLKW   1               ; Current SLT/PVC address
        000044                  L$NXT:  .BLKW   1               ; Pointer to next SLT/PVC address
        000046                  L$SNM:  .BLKW   1               ; REMAINING NUMBER OF SYSTEM LINES
        000050                  L$TPT:  .BLKW   1               ; Current tributary pointer (-1 for PSI)
        000052                  L$TNM:  .BLKB   1               ; REMAINING NUMBER OF TRIBUTARIES
        000053                  L$CTB:  .BLKB   1               ; CURRENT TRIBUTARY NUMBER
        000054                  L$CHN:  .BLKB   1               ; Channel number / X.25 port number and
        000055                  L$PDV:  .BLKB   1               ; ... PDV assigned
        000056                  L$MSG:  .BLKW   1               ; ERROR MESSAGE STRING POINTER
        000060                  L$BUF:  .BLKW   1               ; SAVED BUFFER POINTER
        000062                  L$OPT:  .BLKB   1               ; SAVED OPTIONS BYTE
        000063                  L$TYP:  .BLKB   1               ; CIRCUIT-ID FORMAT TYPE
        000064                  L$FLG:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000066                  L$FL1:  .BLKW   1               ; FLAG WORD (FOR COMMAND)
        000070                  L$FL2:  .BLKW   1               ; FLAG WORD (FOR CURRENT CIRCUIT)
        000072                  L$PAR:  .BLKW   1               ; CURRENT PARAMETER TYPE
        000074                  L$SLEN: .BLKB   1               ; Significant length of circuit name
        000075                  L$PRO:                          ; Line protocol
        000075                  L$MTYP: .BLKB   1               ; Network management circuit type
        000076                  L$FLX:  .BLKW   1               ; Flags word for X.25 circuit commands
        000100                  L$STA:  .BLKB   1               ; CIRCUIT STATE
        000101                  L$COS:  .BLKB   1               ; CIRCUIT COST
        000102                  L$OWN:  .BLKB   1               ; CIRCUIT OWNER (PDV INDEX)
        000103                  L$TAD:  .BLKB   1               ; TRIBUTARY ADDRESS
        000104                  L$ACT:  .BLKB   1               ; MULTIPOINT ACTIVE RATIO
        000105                  L$DEA:  .BLKB   1               ; MULTIPOINT DEAD RATIO
        000106                  L$DDT:  .BLKW   1               ; DMP DEAD TIMER
        000110                  L$DLT:  .BLKW   1               ; DMP DELAY TIMER
        000112                  L$PLT:  .BLKW   1               ; DMP POLL TIMER
        000114                  L$BBT:  .BLKW   1               ; DMP BABBLE TIMER
        000116                  L$NMT:  .BLKW   1               ; DMP NORMAL TIMER
        000120                  L$XMT:  .BLKW   1               ; DMP TRANSMIT TIMER
        000122                  L$BSA:  .BLKB   1               ; DMP ACTIVE BASE
        000123                  L$BSD:  .BLKB   1               ; DMP DYING BASE
        000124                  L$BSI:  .BLKB   1               ; DMP INACTIVE BASE
```

```
                                        .SBTTL  $ADPVC - Create a new PVC
                                ;+
                                ; *** $ADPVC - Create a new PVC
                                ;
                                ; *** Dummy routine for PSI V2.0 release - trap if logic error forces call.
                                ;
                                ;-
     001222                     $ADPVC::
     001222                             SWSTK$  10$                     ;; This routine should never be called
     001226     000003                  BPT
     001230                     10$:    RETURN

                                        .ENDC   ; NDF R$$11S
                000001                  .END
```

```
**FILE**ID**CIRECH






  CCCCCCCC    IIIIII    RRRRRRRR    EEEEEEEEEE    CCCCCCCC  HH      HH
  CCCCCCCC    IIIIII    RRRRRRRR    EEEEEEEEEE    CCCCCCCC  HH      HH
CC              II      RR      RR  EE          CC          HH      HH
CC              II      RR      RR  EE          CC          HH      HH
CC              II      RR      RR  EE          CC          HH      HH
CC              II      RR      RR  EE          CC          HH      HH
CC              II      RRRRRRRR    EEEEEEEE    CC          HHHHHHHHHH
CC              II      RRRRRRRR    EEEEEEEE    CC          HHHHHHHHHH
CC              II      RR  RR      EE          CC          HH      HH
CC              II      RR  RR      EE          CC          HH      HH
CC              II      RR    RR    EE          CC          HH      HH      ....
CC              II      RR    RR    EE          CC          HH      HH      ....
  CCCCCCCC    IIIIII    RR      RR  EEEEEEEEEE    CCCCCCCC  HH      HH      ....
  CCCCCCCC    IIIIII    RR      RR  EEEEEEEEEE    CCCCCCCC  HH      HH      ....




LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15

```
                                          .SBTTL  MTPCHA - Get multpoint characteristics
                                  ;+
                                  ; **-MTPCHA- Get multipoint characteristics
                                  ;
                                  ; INPUTS          R0 = SLT pointer
                                  ;                 R1 = trib flags pointer
                                  ;                 R3 = address of next byte in message
                                  ;                 R4 = context area pointer
                                  ;
                                  ; OUTPUT          R3 = address of next byte in message
                                  ;
                                  ;-
001164                            MTPCHA:
001164                                    SWSTK$  30$                         ;; ENTER SYSTEM STATE
001170                                    SAVMAP                              ;; SAVE CURRENT MAPPING
001174                                    CALL    FNDDCP                      ;; FIND THE DDCMP LINE TABLE
001200  103404                            BCS     10$                         ;; IF CS, COULDN'T FIND IT
001202  116166  000034  000002G           MOVB    S.STPN(R1),2+R$R2(SP)       ;; RETURN TRIB # IN SAVED R2
001210  000417                            BR      25$                         ;; AND RETURN
001212                            10$:    CALL    FNDPCL                      ;; ELSE, MAYBE IT'S A PCL ???
001216  103404                            BCS     15$                         ;; IF CS, NOPE - NOTHING TO RETURN
001220  116166  000003  000002G           MOVB    S.PSA(R1),2+R$R2(SP)        ;; RETURN PHYSICAL STATION ADDRESS IN SAVED R2
001226  000410                            BR      25$                         ;; AND RETURN
001230                            15$:    CALL    DMPCTB                      ;; CHECK FOR DMP
001234  103005                            BCC     25$                         ;; BR IF FOUND IT
001236                            20$:    RETC    R0                          ;; SET USER C-BIT
001250                            25$:    RESMAP                              ;; RESTORE PREVIOUS MAPPING
001254                                    RETURN

001256  103406                    30$:    BCS     35$                         ; IF CS, NO TRIB TO BE HAD
001260  004567  000000G                   JSR     R5,SETPAR                   ; SET TRIBUTARY NUMBER PARAMETER
001264  002164                            .WORD   MP$TRI                      ; ...
001266  000001                            .WORD   1                           ; ONE BYTE
001270  110223                            MOVB    R2,(R3)+                    ; SET PARAMETER IN BUFFER
001272  000421                            BR      60$                         ; DONE FOR TRIBS
                                  ;
                                  ; RETURN MULTIPOINT ACTIVE RATIO IF IT APPLIES
                                  ;
001274                            35$:    SWSTK$  60$                         ;; ENTER SYSTEM STATE
001300                                    SAVMAP                              ;; SAVE CURRENT MAPPING
001304                                    CALL    FNDDCP                      ;; FIND DDCMP'S LINE TABLE
001310  103406                            BCS     55$                         ;; IF CS, COULDN'T - RETURN
                                                                              ;; Add on Mulitpoint Active
001312  004567  000000G                   JSR     R5,SETPAR                   ;; SET MULTIPOINT ACTIVE PARAMETER
001316  004420                            .WORD   MP$MAC                      ;; ...
001320  000001                            .WORD   1                           ;; ONE BYTE DATA
001322  116123  000010                    MOVB    S.PLA(R1),(R3)+             ;; SET PARAMETER IN MESSAGE
001326                            55$:    RESMAP                              ;; RESTORE PREVIOUS MAPPING
001332  010366  000000G                   MOV     R3,R$R3(SP)                 ;; RETURN R3 IN SAVED R3
001336                            60$:    RETURN

                                          .ENDC   ; NDF S$$BAS & R$$RTR & R$$PRO
```

```
Symbol table

AT$ACL= 000100      DF.UNK= 000100      E$NSEG  000010      KISAR6= ****** GX   LP$WCN= 000040
AT$ADN= 000040      DMPCTB= ****** GX   E$NTIM  000046      K$$CNT= 177546      LP$WDV= 000020
AT$AUP= 000020      D$AMXC  000072      E$NUSE  000004      K$$CSR= 177546      LP$WLD= 000360
AT$CYC= 000004      D$AMXH  000074      E$STRT  000006      K$$LDC= 000000      LP$WTR= 000200
AT$LV1= 000002      D$ANN   000000      E$$XPR= 000000      K$$TPS= 000074      LP$WUN= 000100
AT$LV2= 000001      D$BRPR  000102      FLAG    000020R     LC$EXT  000000      LTIM    000014R
AT$NEX= 000200      D$BRTM  000100      FL.INC= 000004      LC$INT  000001      L$ADJ   000110
AT$UP = 000010      D$DELF  000045      FL.PVC= 000002      LD$LP = 000000      L$BUF   000060
A$CIR   000003      D$DELW  000046      FL.SVC= 000001      LE.NRH= 000200      L$CHN   000054
A$LEN   000004      D$END = 000104      FMTOWN  001340R     LE.NRO= 000001      L$COU   000114
A$NID   000000      D$FNB   000034      FNDCTR  002432R     LE.NSH= 000002      L$CTB   000053
A$TM    000003      D$HIOR  000024      FNDDCP= ****** GX   LE.NSO= 000020      L$CTL   000002
A$TMI   000002      D$HOST  000022      FNDHTM  001402R     LE.RCO= 000040      L$DDM   000000
A$TSZ   000000      D$INAC  000044      FNDLTM  001474R     LE.SAE= 000004      L$FLG   000064
A$TYP   000002      D$INCT  000042      FNDPCL= ****** GX   LE.STT= 000010      L$FLX   000070
A$$CHK= 000000      D$IPL   000051      FNDPLD  001064R     LE.XTU= 000100      L$LTM   000070
A$$CPS= 000000      D$LID   000020      F$$LVL= 000001      LF$END= 000020      L$MSG   000056
A$$PRI= 000000      D$LNAM  000006      GETADJ  002152R     LF$MLT= 040000      L$MTYP  000067
A$$TRP= 000000      D$LNUM  000014      G$$TPP= 000000      LF$REA= 000001      L$NAM   000000
BCDASC= ****** GX   D$LST   000047      G$$TSS= 000000      LF$SEG= 100000      L$NLEN= 000020
BF.TRC= 000004      D$MAXC  000064      C$$TTK= 000000      LF$SIG= 000040      L$NOD   000112
BIT0  = 000001      D$MAXH  000066      G$$WRD= 000000      LF$SKP= 000004      L$NXT   000044
BIT1  = 000002      D$MAXV  000070      HF$DLM= 000002      LF$VR2= 000010      L$OPT   000062
BIT10 = 002000      D$MLL   000040      HF$GWY= 000010      LF$ZER= 000002      L$PAR   000120
BIT11 = 004000      D$MNOD  000041      HF$HOS= 000004      LF.ACT= 100000      L$PDV   000055
BIT12 = 010000      D$NA    000062      HF$XDF= 000020      LF.BRO= 000400      L$PFG   000041
BIT13 = 020000      D$NBEA  000056      HTIM    000012R     LF.BWT= 000007      L$PLB   000116
BIT14 = 040000      D$NBRA  000054      H$CUG   000010      LF.ENA= 002000      L$PRO   000067
BIT15 = 100000      D$NEND= 000054      H$DST   000012      LF.LPB= 001000      L$PVC   000004
BIT2  = 000004      D$NLN   000030      H$D29   000014      LF.MDC= 000100      L$SCN   000020
BIT3  = 000010      D$NN    000060      H$FLG   000000      LF.MFL= 004000      L$SCR   000072
BIT4  = 000020      D$OUTT  000043      H$GLEN  000104      LF.MTP= 000020      L$SLEN  000066
BIT5  = 000040      D$RETF  000050      H$GLT   000044      LF.PAC= 000200      L$SLT   000042
BIT6  = 000100      D$RNN   000002      H$GNAM  000050      LF.RDY= 040000      L$SNM   000046
BIT7  = 000200      D$RTMR  000076      H$GNML= 000020      LF.REA= 010000      L$SYL   000122
BIT8  = 000400      D$SEG   000036      H$GPT   000046      LF.SER= 000040      L$TNM   000052
BIT9  = 001000      D$SER   000032      H$HITS  000034      LF.TIM= 000010      L$TPT   000050
BLKING  000016R     D$SQRL  000052      H$HLEN  000044      LF.UNL= 020000      L$TRB   000040
B$10    000003      D$$BUG= 177514      H$LBDA  000070      LF.X2P= 000000      L$TSZ   000070
B$11    000004      D$$ISK= 000000      H$LBDN  000072      LN.CLO= 000000      L$TYP   000063
B$12    000005      D$$L11= 000001      H$LDTE  000002      LN.DUM= 000005      L$UNT   000003
B$5     000000      D$$YNC= 000000      H$LEN   000042      LN.LOA= 000004      L$XLEN  000124
B$6     000001      D$$YNM= 000000      H$LOTS  000032      LN.LOO= 000003      L$$ASG= 000000
B$7     000002      E$NBR   000014      H$NETW  000024      LN.OAU= 000003      L$$DRV= 000000
B$7S10  000006      E$NBS   000020      H$NML = 000006      LN.OFF= 000001      L$$P11= 000001
CF$VR2= ****** GX   E$NCR   000034      H$NPT   000022      LN.ON = 000000      L$$11R= 000000
CF$VR3= ****** GX   E$NCS   000036      H$PTB   000020      LN.OOP= 000004      L.BABL  000062
CIRECH  000022RG    E$NIC   000044      H$PVC   000006      LN.OPE= 000001      L.BUFU  000052
CURCTX= ****** GX   E$NLEN  000050      H$RDTE  000004      LN.REF= 000002      L.CHA   000034
C$FLAG= ****** GX   E$NLLA  000012      H$RNW   000042      LN.SER= 000002      L.CNTL  000024
C$$CKP= 000000      E$NLNK  000000      H$SVC   000036      LN.STA= 000017      L.COST  000015
C$$ORE= 000400      E$NML   000040      H$TRB   000016      LN.SUB= 000360      L.CRC   000030
C$$RSH= 177564      E$NMR   000024      H$XAVL  000100      LN.TRI= 000006      L.CRS   000040
DECPT = ****** GX   E$NMS   000030      H$XBIA  000074      LP$MPT= 000010      L.CTL   000012
DF.CRC= 000010      E$NNOD  000002      H$X29C  000040      LP$MUX= 000004      L.CVA   177776
DF.FOC= 000020      E$NRT   000042      I$$RAR= 000000      LP$TRB= 000002      L.DDM   000002
DF.RAN= 000040      E$NRTP  000005      I$$RDN= 000000      LP$UNT= 000001      L.DDS   000004
```

```
                                                                     C 14
CIRECH      CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 8

SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

T$NRUL     000007           #6-63
T$NVR      000001           #6-63
T$RPRI     000040           #6-63
T$SVC      000034           #6-63       17-778      17-779      17-795
T$T5       000030           #6-63       13-478
T$T6       000032           #6-63
T.ACT      000031           #6-74
T.CHA      000032           #6-74
T.CSR      000022           #6-74
T.ITM      000001           #6-74
T.QUE      000024           #6-74       6-74
T.QU2    = 000026           #6-74
T.RTY      000030           #6-74
T.TMR      000000           #6-74
X$$MCB   = ******            6-69       6-69
X$$TLC   = 000001           #4-2        17-754
ZF.COU   = 001000           #6-69
ZF.DDM   = 000001           #6-69
ZF.DIA   = 004000           #6-69
ZF.DLC   = 000002           #6-69
ZF.DVP   = 100000           #6-69
ZF.INI   = 040000           #6-69
ZF.KMX   = 000020           #6-69
ZF.LLC   = 000004           #6-69
ZF.LMC   = 000100           #6-69
ZF.MAN   = 020000           #6-69
ZF.MFL   = 000010           #6-69
ZF.MTM   = 000400           #6-69
ZF.MUX   = 000040           #6-69
ZF.PSE   = 002000           #6-69
ZF.SLI   = 010000           #6-69
ZF.TIM   = 000200           #6-69
ZF.X3P   = 000000           #6-69
ZS.ASN   = 100000           #6-69
ZS.BSY   = 140000           #6-69
Z.AVL      000014           #6-69
Z.DAT      000016           #6-69       10-357      14-523      17-751
Z.DSP      000000           #6-69       6-69        14-521      14-530      17-750
Z.FLG      000010           #6-69
Z.LEN    = 000016           #6-69
Z.LLN      000006           #6-69
Z.MAP      000020           #6-69
Z.NAM      000004           #6-69       9-255       17-740
Z.PCB      000012           #6-69
Z.SCH      000007           #6-69
$C5TA    = ******   GX       12-445
$HEADR   = ******   GX       10-366     11-396      13-476      14-551      14-578
```

```
                                                                     D 14
CIRECH      CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 9
```

```
    000134                    30$:    RETURN

            000001                    .END
```

```
                                                                    C 16
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17  Page 5
MACRO CALLS AND LOCAL DATA

     55                                          .SBTTL  MACRO CALLS AND LOCAL DATA
     56                                   ;
     57                                   ; MACRO LIBRARY CALLS
     58                                   ;
     59                                          .MCALL  MANDF$,SAVRG,RESRG,CIRCX$,UCBDF$,NSFDF$,HDRDF$
     60                                          .MCALL  SAVMAP,RESMAP
     61
     62 000000                                   CIRCX$                          ; DEFINE CONTEXT AREA
     63 000000                                   HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     64 000000                                   MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     65 000000                                   NSFDF$                          ; DEFINE NS: FEATURES BITS
     66 000000                                   UCBDF$                          ; DEFINE UCB OFFSETS
```

```
                                                                    D 16
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17  Page 6
```

```
                                                  D 1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-4
Symbol table

MP$SVR   000163           MS$TRI   000005           MX$LIN   000020           PNAERR   000462R          STH3     001434R   002
MP$SWI   002260           MS$UNR   000005           MX$LON   000006           PVAERR   000470R          SXMT     001172R   002
MP$SYP   002424           MS$VMS   000004           MX$NMS   000454           P$$P45=  000000           S$$WRG=  000000
MP$TFL   000156           MS.DON   177600           MX$NOD   000006           P$$WRD=  000000           S$$YSZ=  007600
MP$TH1   002206           MS.MOR   000002           MX$OBJ   000006           Q$$OPT=  000010           S.COST   000001
MP$TH2   002205           MS.PAR   000003           MX$OWN   000040           RETACT   001066R          S.FLG    000000
MP$TH3   002202           MS.SUC   000001           MX$PAR   000036           R$PC  =  ****** GX        S.LEN    000004
MP$TLN   000202           MT$AR4   000003           MX$PAS   000010           R$$DER=  000000           S.NMST   000002
MP$TLO   000172           MT$ASC   000100           MX$RAC   000047           R$$K11=  000001           S.OWNR   000003
MP$TPA   004375           MT$BIL   000001           MX$RID   000047           R$$SND=  000000           S1.DEC=  001000
MP$TRI   002164           MT$BYS   000011           MX$RPS   000047           R$$11M=  000000           S1.DPR=  000400
MP$TST   000144           MT$CI    000007           MX$SID   000040           SACT     001022R   002    S1.DSI=  004000
MP$TYP   002130           MT$CLE   000077           MX$SNK   000377           SALL     000462R   002    S1.ESC=  000002
MP$UCS   004407           MT$COD   000200           MX$TYP   000077           SALL1    000466R   002    S1.IBF=  002000
MP$USR   001750           MT$CON   000001           MX$UID   000020           SBBT     001126R   002    S1.IBY=  000200
MP$VEC   004410           MT$DMC   000004           M$$CRB=  000124           SBLK     001500R   002    S1.OBY=  000100
MP$VER   004406           MT$ETH   000006           M$$CRX=  000000           SBSA     001214R   002    S1.PTH=  000010
MP$WDF   002165           MT$HEX   000040           M$$FCS=  000000           SBSD     001236R   002    S1.RES=  010000
MP$WMX   002177           MT$LPB   000005           M$$MGE=  000000           SBSI     001260R   002    S1.RNE=  000020
MP$XMT   002166           MT$MAX   000037           M$$NET=  000000           SCHN     001542R   002    S1.RNF=  020000
MP$XPF   000242           MT$MUL   000100           M$$OVR=  000000           SCMB     001564R   002    S1.RSP=  000004
MP$XPL   000244           MT$NLE   000017           NF$ACC=  100000           SCOS     000550R   002    S1.RST=  000001
MP$XPT   000240           MT$NON   000001           NF$EVT=  040000           SCUS     001606R   002    S1.TNE=  040000
MP$XXX   177777           MT$NR4   000005           NF.ACC=  000001           SDTE     001662R   002    S1.TSY=  000040
MP$5ST   004406           MT$NTY   000060           NF.BCP=  000040           SETCIR   000000R   002    S1.USI=  100000
MS$ACT   000001           MT$OCT   000060           NF.END=  000200           SF$ANM   000001           S2.BEL=  020000
MS$ADU   000010           MT$PHA   000002           NF.EVT=  000002           SF$DNM   000002           S2.BRQ=  000020
MS$ALO   000007           MT$POI   000000           NF.LLI=  000004           SF.ACT=  000200           S2.CR =  002000
MS$ASE   000006           MT$QP2   000010           NF.LV2=  000100           SF.ENA=  000100           S2.CTO=  040000
MS$ATR   000011           MT$ROU   000000           NF.SLI=  000400           SF.LPB=  000004           S2.CTS=  100000
MS$AUT   000000           MT$RO4   000004           NF.SMC=  000020           SF.MFL=  000040           S2.ELF=  001000
MS$CLE   000003           MT$SEC   000000           NF.SWP=  000010           SF.PAC=  000020           S2.FLF=  000400
MS$DED   000004           MT$SGD   000020           NMTACT   001222R          SF.REA=  000010           S2.IRQ=  000200
MS$DIE   000003           MT$SYS   000002           NM.CLN   003400           SF.SER=  000001           S2.OBF=  004000
MS$DUM   000004           MT$TER   000001           NM.INI   002400           SF.SVC=  000002           S2.ORQ=  000100
MS$FAI   000013           MT$TRI   000002           NM.OPR   003000           SF.UNL=  000040           S2.PCU=  010000
MS$GLO   000000           MT$TYP   007777           NM.VR2   002401           SHTM     000654R   002    S2.RCU=  000001
MS$HOL   000002           MT$USD   000000           NM.VR3   002402           SINA     001302R   002    S2.SRQ=  000040
MS$INA   000002           MT$X25   000003           NODACT   000230R          SIND     001324R   002    S2.WAL=  000010
MS$LOA   000003           MU$INC   000001           NODNAM   000774R   002    SINI     001346R   002    S2.WRA=  000006
MS$LOO   000002           MU$OUT   000002           NODOWN   000760R   002    SLCT     000626R   002    S2.WRB=  000002
MS$OFF   000001           MU$PER   000000           NUMACT   000774R          SLOC     001042R   002    S3.ACR=  000001
MS$ON    000000           MV$III   000000           NXT$$$=  000000           SLTM     000702R   002    S3.CTC=  000004
MS$PRO   000006           MV$IV    000001           N$$ACC=  000001           SMXB     001456R   002    S3.FDX=  000200
MS$REA   000004           MXBACT   001400R          N$$BUF=  000001           SMXR     001710R   002    S3.ICE=  001000
MS$REF   000001           MXDTE =  000017           N$$LDV=  000001           SMXW     001732R   002    S3.MHE=  000400
MS$RES   000003           MXRACT   001042R          N$$MCP=  000001           SNMT     001150R   002    S3.NEC=  000020
MS$RST   000001           MXWACT   001054R          N$$MLL=  000001           SNUM     001754R   002    S3.PPT=  020000
MS$RSX   000002           MX$ACT   000020           N$$MOV=  000010           SOWN     000720R   002    S3.PTH=  004000
MS$RT    000005           MX$CIR   000020           N$$NCT=  000001           SPLT     001104R   002    S3.RAL=  000010
MS$SER   000002           MX$CLN   000020           N$$PEM=  000001           SRET     002002R   002    S3.RES=  010000
MS$SHU   000002           MX$CNM   000006           OWNACT   000256R          SSER     000570R   002    S3.RUB=  040000
MS$SRV   000007           MX$CON   000006           PAREND   000536R          SSTA     000474R   002    S3.TAB=  000002
MS$STA   000000           MX$DAC   000020           PCKBCD   001104R          STAD     001002R   002    S3.TME=  002000
MS$SYN   000012           MX$DTE   000020           PDVID =  ****** GX        STA$$$=  000000           S3.TSY=  000040
MS$TER   000001           MX$FIL   000034           PDVTA =  ****** GX        STH1     001370R   002    S3.8BC=  000100
MS$TOP   000003           MX$LGC   000777           PLTACT   001176R          STH2     001412R   002    S4.ABD=  100000
```

```
                                                  E 1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-5
Symbol table
```

```
                                                                   D  2
CICHIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:11     PAGE 11
MACRO CROSS REFERENCE                                    CREF   04.00
MACRO NAME      REFERENCES

                 8-208      9-216      9-219      9-226      9-228      9-229      9-231      9-233      9-239      9-241
                 9-246      9-248      9-249      9-254      9-256      9-257      9-262      9-264      9-265      9-270
                 9-272      9-277      9-279      9-280      9-282      9-283      9-285      9-286      9-288      9-293
                 9-295      9-300      9-302      9-307      9-309      9-310      9-311      9-316      9-318      9-323
                 9-325      9-330      9-332      9-337      9-339      9-344      9-346      9-351      9-353      9-358
                 9-360      9-365      9-367      9-372      9-374      9-379      9-381      9-386      9-388      9-393
                 9-395      9-400      9-402      9-407      9-409      9-414      9-416      9-417      9-418      9-423
                 9-425      9-430      9-432      9-437      9-439      9-440      9-441      9-442      9-447      9-449
                 9-450      9-455      9-457      9-462      9-464      9-469      9-471      9-472      9-477      9-479
                 9-484      9-487      9-492      9-497
MANDF$          #5-57       5-62
NSFDF$          #5-59       5-63
NTLDF$          #5-57       5-64
NTRAN$          #8-82
PDVDF$          #5-58       5-65
PVCDF$          #5-57       5-66
RESRG           #5-58       13-680
RETC            #5-58       13-697
RETURN           10-585     12-633     13-698     15-768     16-789     19-869     20-890     21-908     22-935     23-961
                 24-978     25-995     26-1013    27-1054    28-1072    29-1090    30-1108    31-1126    32-1144    33-1162
                 34-1180    35-1198    36-1216    37-1234    38-1252    39-1270    40-1288    41-1306
SAVRG           #5-58       13-675
SLTDF$          #5-58       5-67
SWSTK$           13-688
TYP$$$          #8-88      #8-90      #8-94      #8-97      #8-100     #8-103     #8-106     #8-109     #8-112     #8-115
                #8-118     #8-121     #8-124     #8-127     #8-130     #8-133     #8-136     #8-139     #8-142     #8-145
                #8-148     #8-151     #8-154     #8-157     #8-160     #8-163     #8-166     #8-169     #8-172     #8-175
                #8-178     #8-181     #8-184     #8-187     #8-189     #8-194     #8-197     #8-200     #8-207     #8-208
                #9-216     #9-219     #9-226     #9-228     #9-229     #9-231     #9-233     #9-239     #9-241     #9-246
                #9-248     #9-249     #9-254     #9-256     #9-257     #9-262     #9-264     #9-265     #9-270     #9-272
                #9-277     #9-279     #9-280     #9-282     #9-283     #9-285     #9-286     #9-288     #9-293     #9-295
                #9-300     #9-302     #9-307     #9-309     #9-310     #9-311     #9-316     #9-318     #9-323     #9-325
                #9-330     #9-332     #9-337     #9-339     #9-344     #9-346     #9-351     #9-353     #9-358     #9-360
                #9-365     #9-367     #9-372     #9-374     #9-379     #9-381     #9-386     #9-388     #9-393     #9-395
                #9-400     #9-402     #9-407     #9-409     #9-414     #9-416     #9-417     #9-418     #9-423     #9-425
                #9-430     #9-432     #9-437     #9-439     #9-440     #9-441     #9-442     #9-447     #9-449     #9-450
                #9-455     #9-457     #9-462     #9-464     #9-469     #9-471     #9-472     #9-477     #9-479     #9-484
                #9-487     #9-492     #9-497
UCBDF$          #5-58       5-68
$BYTE           #8-82       9-228      9-229      9-231      9-248      9-249      9-279      9-280      9-288      9-309
                 9-310      9-416      9-417      9-439      9-440      9-441
$EOM            #8-82       8-207      9-216      9-484
$ERROR          #8-82       8-208      9-219      9-487
$IMAGE          #8-82       9-282      9-449      9-471
$MATCH          #8-82       9-256      9-264      9-285      9-318      9-325      9-332      9-339      9-346      9-353
                 9-360      9-367      9-374      9-381      9-388      9-395      9-402      9-409      9-425      9-432
                 9-457      9-464      9-479
$NEXT           #8-82       8-92       8-95       8-98       8-101      8-104      8-107      8-110      8-113      8-116
                 8-119      8-122      8-125      8-128      8-131      8-134      8-137      8-140      8-143      8-146
                 8-149      8-152      8-155      8-158      8-161      8-164      8-167      8-170      8-173      8-176
                 8-179      8-182      8-185      8-188      8-195      8-198      8-201
$NULL           #8-82       8-189      9-233      9-257      9-265      9-272      9-283      9-286      9-311      9-418
```

```
                                                                   E  2
CICHIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:11     PAGE 12
```

```
                                                                D 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 9-2
CICHOP - CIRCUIT CHANGE PARAMETERS OPERATE

    244 000372  005303                          DEC     R3                      ; ELSE, BACK UP BUFFER POINTER
    245 000374  111501                          MOVB    (R5),R1                 ; GET THE STRING LENGTH
    246 000376  112523                  30$:    MOVB    (R5)+,(R3)+             ; SET MESSAGE IN BUFFER
    247 000400  005301                          DEC     R1                      ; ...
    248 000402  002375                          BGE     30$                     ; ...
    249
    250                                         .ENDC   ; NDF R$$PRO
    251
    252 000404  132764  000360  000041  40$:    BITB    #LP$WLD,L$PFG(R4)       ; IS THIS A MULTIPLE CIRCUIT REQUEST ?
    253 000412  001402                          BEQ     50$                     ; IF EQ, NO - DON'T FORMAT CIRCUIT-ID
    254 000414                                  CALL    FMTLIN                  ; ELSE, FORMAT CIRCUIT-ID STRING
    255 000420  010302                  50$:    MOV     R3,R2                   ; SAVE CURRENT PLACE INTO BUFFER
    256 000422  016403  000060                  MOV     L$BUF(R4),R3            ; RESTORE BUFFER POINTER
    257 000426  160302                          SUB     R3,R2                   ; CALC MESSAGE LENGTH
    258 000430  105713                          TSTB    (R3)                    ; WAS OPERATION A SUCCESS ?
    259 000432  100001                          BPL     60$                     ; IF PL, SUCCESS
    260 000434  000261                          SEC                             ; ELSE, FAILURE
    261 000436                          60$:    RETURN
```

```
                                                                E 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 10
DSPTCH - DISPATCH TO PROPER ROUTINE
```

```
                                                                  D  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 3

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE              REFERENCES

LS$CUS  = 000004           #7-73        7-73
LS$DDT  = 001000           #7-73
LS$DEA  = 000100           #7-73
LS$DLM  = 001004           #7-73
LS$DLT  = 002000           #7-73
LS$DTE  = 000010           #7-73        7-73
LS$HBT  = 000400           #7-73
LS$HTM  = 010000           #7-73        8-124
LS$INA  = 000010           #7-73        8-115
LS$IND  = 000020           #7-73        8-116
LS$INI  = 000040           #7-73        8-117
LS$LCT  = 020000           #7-73        7-73
LS$LMB  = 000002           #7-73
LS$LOO  = 040000           #7-73
LS$MDT  = 020000           #7-73
LS$MRT  = 000020           #7-73
LS$MWN  = 000040           #7-73
LS$MXB  = 001000           #7-73        8-121
LS$MXR  = 000020           #7-73        7-73         7-73
LS$MXW  = 000040           #7-73        7-73         7-73         7-73
LS$NMT  = 020000           #7-73        8-100
LS$NOR  = 100000           #7-73
LS$NTI  = 000200           #7-73
LS$NTL  = 000001           #7-73        8-86         12-357       12-359
LS$NUM  = 000100           #7-73        7-73
LS$OWN  = 000010           #7-73        8-89
LS$PLT  = 004000           #7-73
LS$PRO  = 010000           #7-73
LS$PVC  = 020053           #7-73
LS$RET  = 000200           #7-73        7-73         7-73
LS$SER  = 100000           #7-73        8-103
LS$STA  = 000002           #7-73        8-95         11-322
LS$SVC  = 000362           #7-73
LS$TAD  = 000020           #7-73        8-91
LS$TH1  = 000100           #7-73        8-118
LS$TH2  = 000200           #7-73        8-119
LS$TH3  = 000400           #7-73        8-120
LS$XMT  = 040000           #7-73        8-101
LS.AST  = 100000           #6-66
LS.CEX  = 000001           #6-66
LS.CIR  = 000010           #6-66
LS.CXO  = 010000           #6-66
LS.CX5  = 000400           #6-66
LS.DLM  = 004000           #6-66
LS.ECH  = 001000           #6-66
LS.FDX  = 004000           #6-66
LS.HDX  = 010000           #6-66
LS.LIN  = 000003           #6-66
LS.LMC  = 000007           #6-66
LS.NTI  = 004000           #6-66
LS.ON   = 000011           #6-66
LS.OPT  = 000400           #6-66
```

```
                                                                  E  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 4
```

```
        000020                          LP$WDV = 20             ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40             ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100            ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200            ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK

                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                ;

        000001                          LS$NTL=1                ; NTL SET FUNCTION
        000002                          LS$STA=2                ; SET STATE
        000004                          LS$COS=4                ; SET COST
        000010                          LS$OWN=10               ; SET OWNER
        000020                          LS$TAD=20               ; SET TRIBUTARY ADDRESS
        000040                          LS$ACT=40               ; SET MULTIPOINT ACTIVE
        000100                          LS$DEA=100              ; SET MULTIPOINT DEAD
        000200                          LC$NTL=200              ; NTL CLEAR FUNCTION
        000400                          LC$OWN=400              ; CLEAR OWNER
        001000                          LS$DDT=1000             ; SET DEAD TIMER
        002000                          LS$DLT=2000             ; SET DELAY TIMER
        004000                          LS$PLT=4000             ; SET POLL TIMER
        010000                          LS$BBT=10000            ; SET BABBLE TIMER
        020000                          LS$NMT=20000            ; SET NORMAL TIMER
        040000                          LS$XMT=40000            ; SET TRANSMIT TIMER
        100000                          LS$SER=100000           ; SET SERVICE [ENABLE/DISABLE]

                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FL1)
                                ;

        000001                          LS$BSA=1                ; SET ACTIVE BASE
        000002                          LS$BSD=2                ; SET DYING BASE
        000004                          LS$BSI=4                ; SET INACTIVE BASE
        000010                          LS$INA=10               ; SET ACTIVE INCREMENT
        000020                          LS$IND=20               ; SET DYING INCREMENT
        000040                          LS$INI=40               ; SET INACTIVE INCREMENT
        000100                          LS$TH1=100              ; SET DEAD THRESHOLD
        000200                          LS$TH2=200              ; SET DYING THRESHOLD
        000400                          LS$TH3=400              ; SET INACTIVE THRESHOLD
        001000                          LS$MXB=1000             ; SET MAXIMUM BLOCKS
        010000                          LS$HTM=10000            ; SET HELLO TIMER
        020000                          LS$MDT=20000            ; SET MODEM TEST
        040000                          LS$LOO=40000            ; SET CONTROLLER LOOPBACK
        100000                          LS$NOR=100000           ; SET CONTROLLER NORMAL
                                        .IF DF  R$$PRO  ; PRO/DECNET

                                ;
                                ; LOOPBACK TYPE WORD BIT DEFINITIONS (L$LTY)
                                ;

                                        LS$INT = 0              ; INTERNAL LOOPBACK
                                        LS$EXT = 1              ; EXTERNAL LOOPBACK

                                ;
                                ; MODEM TEST TYPE WORD BIT DEFINITIONS (L$MDT)
                                ;

                                        LS$LCL = 0              ; LOCAL MODEM TEST
                                        LS$REM = 1              ; REMOTE MODEM TEST
                                        LS$OFF = 2              ; SHUT OFF MODEM TESTS
                                        .ENDC   ; DF R$$PRO

                                ;
                                ; Flags word bit definitions for X.25 circuits and lines (L$FLX)
                                ;
```

```
    505                                                    .SBTTL  STAON - REQUEST CIRCUIT STATE "ON"
    506                                            ;+
    507                                            ; **-STAON-REQUEST CIRCUIT STATE TO "ON"
    508                                            ;
    509                                            ; THIS ROUTINE IS CALLED TO CHECK THE CURRENT CIRCUIT STATE, AND
    510                                            ; IF LEGAL, CHANGE IT TO ON.
    511                                            ;
    512                                            ; INPUTS:
    513                                            ;       R2 = ADDRESS OF STATE/OWNER WORD IN SLT
    514                                            ;       R4 = ADDRESS OF CONTEXT BLOCK
    515                                            ;
    516                                            ; OUTPUTS:
    517                                            ;       IF CC, SUCCESS
    518                                            ;       ELSE, R0 CONTAINS ERROR CODE
    519                                            ;-
    520
    521 001534                                     STAON:
    522                                            ;
    523                                            ; CHECK FOR ILLEGAL STATES
    524                                            ;
    525 001534  011201                                     MOV     (R2),R1                 ; GET CURRENT STATE AND OWNER
    526 001536  010100                                     MOV     R1,R0                   ; ...
    527 001540  142701  177417                             BICB    #^C360,R1               ; ISOLATE SUB-STATE BITS
    528 001544  140100                                     BICB    R1,R0                   ; ISOLATE STATE BITS
    529 001546  120027  000000                             CMPB    R0,#MS$ON               ; IS THE CIRCUIT ALREADY ON ?
    530 001552  001454                                     BEQ     100$                    ; IF EQ, YES - NO-OP
    531 001554  120027  000001                             CMPB    R0,#MS$OFF              ; ELSE, IS STATE OFF ?
    532 001560  001403                                     BEQ     10$                     ; IF EQ, YES - PROCEED
    533 001562  120027  000002                             CMPB    R0,#MS$SER              ; ELSE, IS IT SERVICE STATE ?
    534 001566  001006                                     BNE     20$                     ; IF NE, NO - ILLEGAL STATE
    535                                            ;
    536                                            ; IF SUB-STATE = REFLECTING, JUST SET STATE TO ON AND RETURN
    537                                            ;
    538 001570  120127  000040                     10$:    CMPB    R1,#<LN.REF*20>         ; IS THE SUB-STATE REFLECTING ?
    539 001574  001003                                     BNE     20$                     ; IF NE, NO - CONTINUE WITH SCAN
    540 001576  112712  000040                             MOVB    #<LN.REF*20!MS$ON>,(R2) ; ELSE, SET NEW STATE TO ON, REFLECTING
    541 001602  000440                                     BR      100$                    ; AND RETURN
    542                                            ;
    543                                            ; MAKE SURE CIRCUIT HAS AN OWNER
    544                                            ;
    545 001604  012700  177765                     20$:    MOV     #ME.CST,R0              ; ASSUME CIRCUIT IS WRONG STATE
    546 001610  012764  000003  000072                     MOV     #ME$CIR,L$PAR(R4)       ; ...
    547 001616  116201  000001                             MOVB    1(R2),R1                ; GET THE CIRCUIT OWNER'S PDV INDEX
    548 001622  001427                                     BEQ     90$                     ; IF EQ, NO OWNER SET - ERROR
    549                                            ;
    550                                            ; REASSIGN THE CIRCUIT TO IT'S TRUE OWNER
    551                                            ;
    552 001624  120164  000055                             CMPB    R1,L$PDV(R4)            ; IS THE OWNER ALREADY CORRECT ?
    553 001630  001412                                     BEQ     30$                     ; IF EQ, YES - SKIP REASSIGN
    554 001632  067701  000000G                            ADD     @PDVTA,R1               ; GET THE OWNER'S PDV ADDRESS   ; RJK08
    555 001636  011101                                     MOV     (R1),R1                 ; POINT TO THE PDV
    556 001640  016101  000004                             MOV     Z.NAM(R1),R1            ; GET THE RAD50 OWNER NAME
    557 001644  012700  017020                             MOV     #<36*400!20>,R0         ; SET REASSIGN FUNCTION CODE
    558 001650                                             CALL    DLXQIO                  ; ISSUE QIO
    559 001654  103404                                     BCS     40$                     ; IF CS, OPERATION FAILED
    560                                            ;
    561                                            ;  TELL DLX TO TURN THE CIRCUIT ON
```

```
000134  012700  017020          30$:    MOV     #<36*400!20>,R0         ; SET REASSIGN FUNCTION CODE
000140  116401  000102                  MOVB    L$OWN(R4),R1            ; GET OWNER PDV ID
000144  067701  000000G                 ADD     @PDVTA,R1               ; POINT TO PDV ADDRESS          ; RJK08
000150  011101                          MOV     (R1),R1                 ; GET PDV ADDRESS
000152  016101  000004                  MOV     Z.NAM(R1),R1            ; GET RAD50 PROCESS NAME
000156                                  CALL    DLXQIO                  ; ISSUE REASSIGN
000162  103413                          BCS     50$                     ; IF CS, OPERATION FAILED
                                ;
                                ; SET NEW OWNER IN SLT
                                ;
000164  105762  000014                  TSTB    L.NSTA(R2)              ; MULTIPOINT CIRCUIT ?
000170  001004                          BNE     40$                     ; IF NE, YES
000172  116462  000102  000021          MOVB    L$OWN(R4),L.NMST+1(R2)  ; ELSE, SET THE OWNER FOR THE CIRCUIT
000200  000407                          BR      70$                     ; AND RETURN
000202  116463  000102  000003  40$:    MOVB    L$OWN(R4),S.NMST+1(R3)  ; SET THE NEW OWNER
000210  000403                          BR      70$                     ;
000212  012700  177747          50$:    MOV     #ME.OPE,R0              ; SET OPERATION FAILED ERROR
000216  000261                  60$:    SEC                             ; FAILURE
000220                          70$:    RETURN

                                        .IF NDF S$$BAS & R$$RTR & R$$PRO
```

```
                                                                   D  8
CICHPD      CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE              REFERENCES

LP$WDV  = 000020            #6-86       6-86
LP$WLD  = 000360            #6-86
LP$WTR  = 000200            #6-86       6-86       9-165      10-201      23-1086     23-1091
LP$WUN  = 000100            #6-86       6-86
LR.AST    000026            #5-71
LR.CIR    000002            #5-71
LR.CTL    000004            #5-71      *11-229
LR.EFN    000004            #5-71
LR.LIN    000002            #5-71      *11-228
LR.PRM    000030            #5-71
LR.PRO    000002            #5-71
LR.STS    000000            #5-71       9-169
LR.TCB    000002            #5-71
LR.UNT    000005            #5-71      *11-230
LS$ACT  = 000040            #6-86
LS$BBT  = 010000            #6-86
LS$BLK  = 001000            #6-86       6-86
LS$BSA  = 000001            #6-86
LS$BSD  = 000002            #6-86
LS$BSI  = 000004            #6-86
LS$CHN  = 000001            #6-86       6-86
LS$CMB  = 000002            #6-86       6-86       6-86       6-86
LS$COS  = 000004            #6-86
LS$CUS  = 000004            #6-86       6-86
LS$DDT  = 001000            #6-86
LS$DEA  = 000100            #6-86
LS$DLM  = 001004            #6-86
LS$DLT  = 002000            #6-86
LS$DTE  = 000010            #6-86       6-86
LS$HBT  = 000400            #6-86
LS$HTM  = 010000            #6-86
LS$INA  = 000010            #6-86
LS$IND  = 000020            #6-86
LS$INI  = 000040            #6-86
LS$LCT  = 020000            #6-86       6-86
LS$LMB  = 000002            #6-86
LS$LOO  = 040000            #6-86
LS$MDT  = 020000            #6-86
LS$MRT  = 000020            #6-86
LS$MWN  = 000040            #6-86
LS$MXB  = 001000            #6-86
LS$MXR  = 000020            #6-86       6-86       6-86
LS$MXW  = 000040            #6-86       6-86       6-86       6-86
LS$NMT  = 020000            #6-86
LS$NOR  = 100000            #6-86
LS$NTI  = 000200            #6-86
LS$NTL  = 000001            #6-86
LS$NUM  = 000100            #6-86       6-86
LS$OWN  = 000010            #6-86
LS$PLT  = 004000            #6-86
LS$PRO  = 010000            #6-86
LS$PVC  = 020053            #6-86
```

```
                                                                   E  8
CICHPD      CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 4
```

```
        000125                          L$INA:  .BLKB   1               ; DMP ACTIVE INCREMENT
        000126                          L$IND:  .BLKB   1               ; DMP DYING INCREMENT
        000127                          L$INI:  .BLKB   1               ; DMP INACTIVE INCREMENT
        000130                          L$TH1:  .BLKB   1               ; DMP DEAD THRESHOLD
        000131                          L$TH2:  .BLKB   1               ; DMP DYING THRESHOLD
        000132                          L$TH3:  .BLKB   1               ; DMP INACTIVE THRESHOLD
        000133                          L$MXB:  .BLKB   1               ; DMP MAXIMUM BLOCKS
        000134                          L$NTL:  .BLKW   13.             ; NTL MESSAGE BUFFER
        000166                          L$SCR:  .BLKW   15.             ; SCRATCH BUFFER
        000224                          L$LCT:  .BLKW   1               ; COUNTER TIMER
        000226                          L$HTM:  .BLKW   1               ; HELLO TIMER
        000230                          L$SER:                          ; SERVICE
        000230                          L$LTM:  .BLKW   1               ; LISTEN TIMER
        000232                          L$XCH:  .BLKW   1               ; X25 Logical Channel Number (LCN)
        000234                          L$LMB:                          ; X25 Max Block (Line)
        000234                          L$CMB:  .BLKW   1               ; X25 Max Data (Circuit)
        000236                          L$NUML:                         ; DLM Number length
        000236                          L$DTEL: .BLKW   1               ; X25 DTE length (Circuit)
        000240                          L$NUM:                          ; DLM Number
        000240                          L$DTE:  .BLKB   8.              ; X25 DTE (Circuit)
        000250                          L$DTEP: .BLKW   1               ; X25 Pointer to DTE descriptor (Circuit)
        000252                          L$MWN:                          ; X25 Max Window (Line)
        000252                          L$MXW:  .BLKB   1               ; X25 Max Window (Circuit)
        000253                          L$MRT:                          ; X25 Max Retransmits (Line)
        000253                          L$MXR:  .BLKB   1               ; DLM Max Recalls (Circuit)
        000254                          L$HBT:  .BLKW   1               ; X25 Holdback Timer (Line)
        000256                          L$NTI:                          ; X25 Retransmit Timer (Line)
        000256                          L$RET:  .BLKW   1               ; DLM Recall Timer (Circuit)
        000260                          L$CUS:  .BLKW   1               ; DLM Usage
        000262                          L$BLK:  .BLKW   1               ; DLM blocking state
                                                .IF DF  R$$PRO  ;PRO/DECnet
                                        L$LTY:  .BLKW   1               ; Loopback Type
                                        L$MDT:  .BLKW   1               ; Modem Test
                                                .ENDC   ; DF R$$PRO
                                                .EVEN
        000264                          L$LEN:                          ; LENGTH OF CONTEXT AREA
        000000                                  .PSECT
                                        ;
                                        ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                        ;
                000001                          LF$REA = 1              ; READ COUNTERS OPERATION
                000002                          LF$ZER = 2              ; ZERO COUNTERS OPERATION
                000004                          LF$SKP = 4              ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
                000010                          LF$VR2 = 10             ; CONNECTED TO VERSION 2.0 NCP
                040000                          LF$MLT  = 40000         ; MULTIPLE ADJACENCY FLAG
                100000                          LF$SEG  = 100000        ; SEGMENTED RESPONSE IN PROGRESS
                                        ;
                                        ; PARSE FLAG DEFINITIONS (L$PFG)
                                        ;
                000001                          LP$UNT = 1              ; UNIT NUMBER FOUND
                000002                          LP$TRB = 2              ; TRIBUTARY NUMBER FOUND
                000004                          LP$MUX = 4              ; DEVICE IS MUX
                000010                          LP$MPT = 10             ; LINE IS MULTIPOINT
```

```
                                                                D 10
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 16-1
Symbol table

A$$CHK= 000000        H$FLG   000000        LF.X2P= 000000        LS$NTL= 000001        L$NMT   000116
A$$CPS= 000000        H$GLEN  000104        LN.CLO= 000000        LS$NUM= 000100        L$NTI   000256
A$$PRI= 000000        H$GLT   000044        LN.DUM= 000005        LS$OWN= 000010        L$NTL   000134
A$$TRP= 000000        H$GNAM  000050        LN.LOA= 000004        LS$PLT= 004000        L$NUM   000240
CF$BLK= 000102        H$GNML= 000020        LN.LOO= 000003        LS$PRO= 010000        L$NUML  000236
CICHPX  000200RG      H$GPT   000046        LN.OAU= 000003        LS$PVC= 020053        L$NXT   000044
CKCSTA  000536R       H$HITS  000034        LN.OFF= 000001        LS$RET= 000200        L$OPT   000062
CKDSVC  001012R       H$HLEN  000044        LN.ON = 000000        LS$SER= 100000        L$OWN   000102
C$DTE   000012        H$LBDA  000070        LN.OOP= 000004        LS$STA= 000002        L$PAR   000072
C$FLG   000015        H$LBDN  000072        LN.OPE= 000001        LS$SVC= 000362        L$PDV   000055
C$GNAM  000016        H$LDTE  000002        LN.REF= 000002        LS$TAD= 000020        L$PFG   000041
C$LCN   000010        H$LEN   000042        LN.SER= 000002        LS$TH1= 000100        L$PLT   000112
C$LEN   000016        H$LOTS  000032        LN.STA= 000017        LS$TH2= 000200        L$PRO   000075
C$LNK   000000        H$NETW  000024        LN.SUB= 000360        LS$TH3= 000400        L$PVC   000004
C$NAM   000002        H$NML = 000006        LN.TRI= 000006        LS$XMT= 040000        L$RET   000256
C$NML   000002        H$NPT   000022        LP$MPT= 000010        L$ACT   000104        L$SCN   000020
C$PORT  000014        H$PTB   000020        LP$MUX= 000004        L$BBT   000114        L$SCR   000166
C$$CKP= 000000        H$PVC   000006        LP$NXC= 100000        L$BLK   000262        L$SER   000230
C$$ORE= 000400        H$RDTE  000004        LP$TRB= 000002        L$BSA   000122        L$SLEN  000074
C$$RSH= 177564        H$RNW   000042        LP$UNT= 000001        L$BSD   000123        L$SLT   000042
D$$BUG= 177514        H$SVC   000036        LP$WCN= 000040        L$BSI   000124        L$SNM   000046
D$$ISK= 000000        H$TRB   000016        LP$WDV= 000020        L$BUF   000060        L$STA   000100
D$$L11= 000001        H$XAVL  000100        LP$WLD= 000360        L$CHN   000054        L$TAD   000103
D$$YNC= 000000        H$XBIA  000074        LP$WTR= 000200        L$CMB   000234        L$TH1   000130
D$$YNM= 000000        H$X29C  000040        LP$WUN= 000100        L$COS   000101        L$TH2   000131
E$NBR   000014        I$$RAR= 000000        LS$ACT= 000040        L$CTB   000053        L$TH3   000132
E$NBS   000020        I$$RDN= 000000        LS$BBT= 010000        L$CTL   000002        L$TNM   000052
E$NCR   000034        KISAR6= ****** GX     LS$BLK= 001000        L$CUS   000260        L$TPT   000050
E$NCS   000036        K$$CNT= 177546        LS$BSA= 000001        L$DDM   000000        L$TRB   000040
E$NIC   000044        K$$CSR= 177546        LS$BSD= 000002        L$DDT   000106        L$TYP   000063
E$NLEN  000050        K$$LDC= 000000        LS$BSI= 000004        L$DEA   000105        L$UNT   000003
E$NLLA  000012        K$$TPS= 000074        LS$CHN= 000001        L$DLT   000110        L$XCH   000232
E$NLNK  000000        LC$EXT  000000        LS$CMB= 000002        L$DTE   000240        L$XMT   000120
E$NML   000040        LC$INT  000001        LS$COS= 000004        L$DTEL  000236        L$$ASG= 000000
E$NMR   000024        LC$NTL= 000200        LS$CUS= 000004        L$DTEP  000250        L$$DRV= 000000
E$NMS   000030        LC$OWN= 000400        LS$DDT= 001000        L$FLG   000064        L$$P11= 000001
E$NNOD  000002        LD$LP = 000000        LS$DEA= 000100        L$FLX   000076        L$$11R= 000000
E$NRT   000042        LF$MLT= 040000        LS$DLM= 001004        L$FL1   000066        L.COST  000015
E$NRTP  000005        LF$REA= 000001        LS$DLT= 002000        L$FL2   000070        L.CTL   000012
E$NSEG  000010        LF$SEG= 100000        LS$DTE= 000010        L$HBT   000254        L.CVA   177776
E$NTIM  000046        LF$SKP= 000004        LS$HBT= 000400        L$HTM   000226        L.DDM   000002
E$NUSE  000004        LF$VR2= 000010        LS$HTM= 010000        L$INA   000125        L.DDS   000004
E$STRT  000006        LF$ZER= 000002        LS$INA= 000010        L$IND   000126        L.DLC   000003
E$$XPR= 000000        LF.ACT= 100000        LS$IND= 000020        L$INI   000127        L.DLM   000006
FNDCTR= ****** GX     LF.BRO= 000400        LS$INI= 000040        L$LCT   000224        L.DLS   000010
F$$LVL= 000001        LF.BWT= 000007        LS$LCT= 020000        L$LEN   000264        L.FLG   000000
G$$TPP= 000000        LF.ENA= 002000        LS$LMB= 000002        L$LMB   000234        L.KRBA  000016
G$$TSS= 000000        LF.LPB= 001000        LS$LOO= 040000        L$LTM   000230        L.LEN = 000022
G$$TTK  000000        LF.MDC= 000100        LS$MDT= 020000        L$MRT   000253        L.MPF   000022
G$$WRD  000000        LF.MFL= 004000        LS$MRT= 000020        L$MSG   000056        L.NMST  000020
HF$DLM  000002        LF.MTP= 000020        LS$MWN= 000040        L$MTYP  000075        L.NSTA  000014
HF$GWY  000010        LF.PAC= 000200        LS$MXB= 001000        L$MWN   000252        L.OWNR  000021
HF$HOS  000004        LF.RDY= 040000        LS$MXR= 000020        L$MXB   000133        L.UNT   000013
HF$XDP  000020        LF.REA= 010000        LS$MXW= 000040        L$MXR   000253        MB$DIS  000001
H$CUG   000010        LF.SER= 000040        LS$NMT= 020000        L$MXW   000252        MB$ENA  000000
H$DST   000012        LF.TIM= 000010        LS$NOR= 100000        L$NAM   000000        MB$FUL  000002
H$D29   000014        LF.UNL= 020000        LS$NTI= 000200        L$NLEN= 000020        MB$MIX  000002
```

```
                                                                 E 10
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 16-2
```

Table of contents

```
     6-   55    Macro calls
     7-   81    BIT DEFINITIONS
     8-   83    Local data
     9-  103    CIRECH - Read circuit characteristics
    10-  336    FNDPLD - Find physical link data base
    11-  371    MTPCHA - Get multpoint characteristics
    12-  424    FMTOWN - Format an ASCII owner field
    13-  457    FNDHTM - Find hello timer
    14-  486    FNDLTM - Find adjacent node and listen timer
    15-  600    GETADJ - Get an adjacency
    16-  647    SETNOD - Put node id
    17-  704    FNDCTR -  Find counter block
    18-  800    SVCRCL - Format SVC recall values
    19-  837    SVCBLK - Get SVC block parameters
    20-  888    SETSER - Set Service parameter
    21-  919    SETNAM - Set an image name field
```

```
                                          .SBTTL  FMTOWN - Format an ASCII owner field
                              ;+
                              ; **-FMTOWN-FORMAT AN ASCII OWNER FIELD
                              ;
                              ; THIS ROUTINE TAKES A RAD50 PROCESS NAME AND BUILDS
                              ; AN IMAGE FIELD, DELETING TRAILING SPACES.
                              ;
                              ; INPUTS:
                              ;         R1 = RAD50 PROCESS NAME
                              ;         R3 = ADDRESS OF BUFFER FOR IMAGE STRING
                              ;
                              ; OUTPUTS:
                              ;         R3 = NEXT FREE BYTE IN BUFFER
                              ;
                              ; REGISTERS:
                              ;         R4, R5 ARE PRESERVED
                              ;-

001340                        FMTOWN:
001340  105723                          TSTB    (R3)+                   ; SAVE SPACE FOR IMAGE COUNT
001342  010300                          MOV     R3,R0                   ; COPY BUFFER POINTER
001344                                  CALL    $C5TA                   ; CONVERT NAME TO ASCII
001350  124027  000040                  CMPB    -(R0),#'                ; WAS THE LAST CHAR A SPACE ?
001354  001004                          BNE     10$                     ; IF NE, NO
001356  124027  000040                  CMPB    -(R0),#'                ; ...
001362  001001                          BNE     10$                     ; ...
001364  005300                          DEC     R0                      ; ...
001366  005200                10$:      INC     R0                      ; POINT TO NEXT FREE SPACE
001370  160300                          SUB     R3,R0                   ; CALC LENGTH OF NAME
001372  110063  177777                  MOVB    R0,-1(R3)               ; SET LENGTH OF NAME IN FIELD
001376  060003                          ADD     R0,R3                   ; POINT TO NEXT FREE BYTE IN BUFFER
001400                                  RETURN
```

```
                                                                     D 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-2
Symbol table

L.DISL   000042          MC$CLD   001464          MC$TIM   000000          MD$KDZ   000036          ME.FOP   177763
L.DISR   000044          MC$COU   100000          MC$TYP   007777          MD$KL    000040          ME.FOR   177776
L.DLC    000003          MC$CTL   001454          MC$UBU   002052          MD$KMX   000054          ME.FUN   177777
L.DLM    000006          MC$CTR   001452          MC$UFD   002047          MD$KMY   000052          ME.GRO   177745
L.DLS    000010          MC$CTS   001453          MC$UMR   001132          MD$LEN   000050          ME.HAR   177750
L.ENA    000066          MC$DOV   002050          MC$UMS   001133          MD$PCL   000011          ME.IID   177767
L.FLG    000000          MC$LBR   001750          MC$VAX   000003          MD$QNA   000005          ME.LCO   177766
L.KRBA   000016          MC$LBS   001751          MC$WID   060000          MD$UNA   000001          ME.LPR   177757
L.LEN =  000022          MC$LDI   001774          MC$WIH   040000          MD$WIT   000002          ME.MPR   177773
L.LNUM   000046          MC$LDO   001775          MC$WIL   020000          MD.ABO   000017          ME.MVE   177771
L.MPF    000022          MC$LDR   001762          MC$W08   020000          MD.ACC   000010          ME.OPE   177747
L.NAST   000065          MC$LDS   001763          MC$W16   040000          MD.AOB   000016          ME.PLO   177751
L.NCUB   000012          MC$LLB   002021          MC$W32   060000          MD.BOB   000011          ME.PMI   177743
L.NCUL=  000050          MC$LLP   002115          MC$XBR   001750          MD.DIA   000007          ME.PNA   177752
L.NIP    000010          MC$LLR   002007          MC$XBS   001751          MD.DOB   000015          ME.PRI   177775
L.NLSE   000054          MC$LOD   000001          MC$XCJ   000322          MD.DUM   000002          ME.PTY   177772
L.NMST   000020          MC$LRB   002020          MC$XCR   002260          MD.FAI   000014          ME.PVA   177760
L.NRSE   000053          MC$LRP   002114          MC$XCS   002261          MD.FNA   000001          ME.RES   177761
L.NSTA   000014          MC$LRR   002006          MC$XDR   001762          MD.FOB   000006          ME.ROO   177754
L.OWNR   000021          MC$LSI   002032          MC$XDS   001763          MD.LOA   000001          ME.SIZ   177774
L.PAIC   000017          MC$LST   002033          MC$XFR   002272          MD.LSH   000020          ME.SYS   177746
L.PAIR   000016          MC$LUP   006072          MC$XFS   002273          MD.NOB   000012          ME.UCO   177770
L.PLD    000047          MC$MAP   010000          MC$XLJ   000323          MD.NON   177777          MF$ACT   177776
L.PLL    000012          MC$MBL   001764          MC$XLR   002330          MD.PER   000000          MF$ADD   000000
L.SCFW   000056          MC$MBX   004420          MC$XMA   000310          MD.RES   000004          MF$ADJ   177774
L.STA    000063          MC$MBY   001752          MC$XMC   002305          MD.ROB   000005          MF$ALL   177775
L.STBL   000070          MC$MPX   004406          MC$XMS   002304          MD.RSH   000013          MF$BYE   000300
L.STRC   000050          MC$NAP   001604          MC$XNR   002332          MD.SDU   000005          MF$CHA   000023
L.STRM   000051          MC$NBR   001130          MC$XRC   002316          MD.SLO   000003          MF$DUM   000020
L.STS    000002          MC$NBS   001131          MC$XRR   002331          MD.SNA   000000          MF$EVT   000001
L.ST2    000003          MC$NCR   001154          MC$XRS   002342          MD.TLO   000004          MF$FUP   000264
L.TBP    000004          MC$NCS   001155          MC$020   000002          MD.UNA   000002          MF$KNO   177777
L.TIMC   000060          MC$NML   001274          MD$CI    000007          MD.UNR   000003          MF$LOA   000017
L.TIMI   000001          MC$NMR   001142          MD$CNA   000003          MD.UOB   000007          MF$LOO   177775
L.TIMR   000000          MC$NMS   001143          MD$COU   000001          MD.VOL   000006          MF$REA   000024
L.TOR    000064          MC$NNO   001606          MD$DA    000010          ME$ALI   000010          MF$SIG   177773
L.UNT    000013          MC$NNU   001605          MD$DL    000004          ME$AL2   000003          MF$SPF   000302
L.XSET   000020          MC$NOP   001607          MD$DLV   000020          ME$ARE   000005          MF$SYS   000026
MAPADD=  ****** GX       MC$NOR   000000          MD$DMC   000014          ME$CIR   000003          MF$TES   000022
MB$DIS   000001          MC$NPF   001616          MD$DMF   000046          ME$EXA   000000          MF$TRA   000301
MB$ENA   000000          MC$NPR   001630          MD$DMP   000022          ME$EXE   000200          MF$TRI   000021
MB$FUL   000002          MC$NRE   001200          MD$DMR   000050          ME$LIN   000001          MF$ZER   000025
MB$MIX   000002          MC$NRT   001166          MD$DMV   000042          ME$LOG   000002          ML$ALL   100000
MB$ONE   000001          MC$NVR   001642          MD$DN    000016          ME$MOD   000004          ML$CLS   000000
MB$RXO   000001          MC$P08   000000          MD$DP    000000          ME$NOD   000000          ML$CON   000001
MB$TXO   000000          MC$P11   000001          MD$DPV   000044          ME$NON   177777          ML$EXT   000001
MB$ZER   000000          MC$RFL   002046          MD$DQ    000006          ME$OBJ   000007          ML$FIL   000002
MC$BAB   006073          MC$SBU   002051          MD$DTE   000024          ME$OB2   000004          ML$FIR   000000
MC$BID   001765          MC$SFL   002044          MD$DU    000002          ME$PRO   000005          ML$INT   000000
MC$BMC   001767          MC$SYC   000012          MD$DUP   000012          ME$SYS   000006          ML$KNO   140000
MC$BSC   001766          MC$SYL   000036          MD$DV    000026          ME.BLO   177744          ML$MON   000003
MC$CAC   001442          MC$SYR   000050          MD$DZ    000030          ME.CON   177753          ML$SYS   000002
MC$CAP   001440          MC$SYS   000024          MD$FUL   000000          ME.CST   177765          ML$TOP   000001
MC$CCL   001445          MC$SYZ   000000          MD$HAL   000001          ME.DIS   177755          MN$UNL   000377
MC$CDC   002045          MC$TBR   001140          MD$HEL   000002          ME.DON   177600          MO$ACC   000200
MC$CDP   001441          MC$TBS   001141          MD$KCP   000013          ME.FCO   177762          MO$ADD   000002
MC$CIF   001465          MC$TDR   002053          MD$KDP   000034          ME.FIO   177756          MO$ALA   000004
```

```
                                                                     E 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-3
```

```
MACRO CROSS REFERENCE                                       CREF    04.00

MACRO NAME           REFERENCES

ADJDF$          #6-61      6-64
ASL$            #6-61      14-588
BIAS            #6-61      14-532    14-537    17-765    17-782
CALL             9-146     9-151     9-154     9-162     9-194     9-208     9-237     9-257     9-307     9-322
                 10-349    10-351    11-384    11-386    11-390    11-394    11-409    11-411    12-445    13-472
                 13-474    14-508    14-512    14-515    14-549    14-565    14-572    16-689    16-699    18-817
                 19-853    19-867
CALLR            9-131     9-147     9-311
CEACC$          #6-60
CIRCX$          #6-61      6-81
CTRDF$          #6-61      6-63
DDCDF$          #6-60      6-72
DHBDF$          #6-61      6-65
ECDDB$          #6-60      6-67
MANDF$          #6-59      6-66
MAP             #6-60      14-530    15-624    15-639    17-750    17-760    17-785    18-819    19-855
MESAG$          #6-59
PCBDF$          #6-59      6-74
PDVDF$          #6-59      6-69
PHBDF$          #6-59      6-73
PLBDF$          #6-61      6-75
RESMAP          #6-60      11-397    11-418    13-483    15-643    17-791    18-833    19-883
RESRG           #6-59      9-258     14-596    16-701
RETC            #6-60      10-366    11-396    13-476    14-551    14-578
RETURN           9-334     10-367    11-398    11-420    12-455    13-484    14-554    14-582    14-597    15-644
                 17-798    18-835    19-885    20-917    21-950
SAVMAP          #6-60      11-385    11-410    13-473    15-623    17-783    18-818    19-854
SAVRG           #6-59      9-254     14-507    16-661
SLTDF$          #6-59      6-70
SVCDF$          #6-61      6-71
SWSTK$           9-146     10-349    11-384    11-409    13-472    14-512    18-817    19-853
TMPDF$          #6-74      6-74
XPDDB$          #6-61      6-68
.ADDRB          #6-74
.ADDRW          #6-74
.APR            #6-74
.BIN            #6-74
.CNB            #6-74
.CNW            #6-74
.CORE           #6-74
.CSR            #6-74      6-74      6-74
.CTIM           #6-74
.DPRB           #6-74
.DPRW           #6-74
.DVCHA          #6-74      6-74
.INT            #6-74      6-74      6-74
.INT1           #6-74
.INT2           #6-74
.INT3           #6-74
.LFLHD          #6-74
.LIBR           #6-74
.LINKS          #6-74
```

```
                                                                    D 15
CIRECO - READ/AND OR ZERO CIRCU MACRO V05.03b Saturday 29-Jun-85 12:17  Page 10-2
Symbol table

A$$CHK= 000000          LF$MLT= 040000      L$MTYP  000067      MC$CAP  001440      MC$SYS  000024
A$$CPS= 000000          LF$REA= 000001      L$NAM   000000      MC$CCL  001445      MC$SYZ  000000
A$$PRI= 000000          LF$SEG= 100000      L$NLEN= 000020      MC$CDC  002045      MC$TBR  001140
A$$TRP= 000000          LF$SIG= 000040      L$NOD   000112      MC$CDP  001441      MC$TBS  001141
BIT0  = 000001          LF$SKP= 000004      L$NXT   000044      MC$CIF  001465      MC$TDR  002053
BIT1  = 000002          LF$VR2= 000010      L$OPT   000062      MC$CLD  001464      MC$TIM  000000
BIT10 = 002000          LF$ZER= 000002      L$PAR   000120      MC$COU  100000      MC$TYP  007777
BIT11 = 004000          LF.ACT= 100000      L$PDV   000055      MC$CTL  001454      MC$UBU  002052
BIT12 = 010000          LF.BRO= 000400      L$PFG   000041      MC$CTR  001452      MC$UFD  002047
BIT13 = 020000          LF.BWT= 000007      L$PLB   000116      MC$CTS  001453      MC$UMR  001132
BIT14 = 040000          LF.ENA= 002000      L$PRO   000067      MC$DDV  002050      MC$UMS  001133
BIT15 = 100000          LF.LPB= 001000      L$PVC   000004      MC$LBR  001750      MC$VAX  000003
BIT2  = 000004          LF.MDC= 000100      L$SCN   000020      MC$LBS  001751      MC$WID  060000
BIT3  = 000010          LF.MFL= 004000      L$SCR   000072      MC$LDI  001774      MC$WIH  040000
BIT4  = 000020          LF.MTP= 000020      L$SLEN  000066      MC$LDO  001775      MC$WIL  020000
BIT5  = 000040          LF.PAC= 000200      L$SLT   000042      MC$LDR  001762      MC$W08  020000
BIT6  = 000100          LF.RDY= 040000      L$SNM   000046      MC$LDS  001763      MC$W16  040000
BIT7  = 000200          LF.REA= 010000      L$SYL   000122      MC$LLB  002021      MC$W32  060000
BIT8  = 000400          LF.SER= 000040      L$TNM   000052      MC$LLP  002115      MC$XBR  001750
BIT9  = 001000          LF.TIM= 000010      L$TPT   000050      MC$LLR  002007      MC$XBS  001751
B$10    000003          LF.UNL= 020000      L$TRB   000040      MC$LOD  000001      MC$XCJ  000322
B$11    000004          LF.X2P= 000000      L$TSZ   000070      MC$LRB  002020      MC$XCR  002260
B$12    000005          LN.CLD= 000000      L$TYP   000063      MC$LRP  002114      MC$XCS  002261
B$5     000000          LN.DUM= 000005      L$UNT   000003      MC$LRR  002006      MC$XDR  001762
B$6     000001          LN.LOA= 000004      L$$ASG= 000000      MC$LSI  002032      MC$XDS  001763
B$7     000002          LN.LOO= 000003      L$$DRV= 000000      MC$LST  002033      MC$XFR  002272
B$7S10  000006          LN.DAU= 000003      L$$P11= 000001      MC$LUP  006072      MC$XFS  002273
CIRECO  000002RG        LN.OFF= 000001      L$$11R= 000000      MC$MAP  010000      MC$XLJ  000323
C$$CKP= 000000          LN.ON  = 000000     L.COST  000015      MC$MBL  001764      MC$XLR  002330
C$$ORE= 000400          LN.OOP= 000004      L.CTL   000012      MC$MBX  004420      MC$XMA  000310
C$$RSH= 177564          LN.OPE= 000001      L.CVA   177776      MC$MBY  001752      MC$XMC  002305
DACIR = ****** GX       LN.REF= 000002      L.DDM   000002      MC$MPX  004406      MC$XMS  002304
DCPCIR= ****** GX       LN.SER= 000002      L.DDS   000004      MC$NAP  001604      MC$XNR  002332
DLCTBL  000000R     002 LN.STA= 000017      L.DLC   000003      MC$NBR  001130      MC$XRC  002316
DLMCOU= ****** GX       LN.SUB= 000360      L.DLM   000006      MC$NBS  001131      MC$XRR  002331
DMPCIR= ****** GX       LN.TRI= 000006      L.DLS   000010      MC$NCR  001154      MC$XRS  002342
D$$BUG= 177514          LP$MPT= 000010      L.FLG   000000      MC$NCS  001155      MC$020  000002
D$$ISK= 000000          LP$MUX= 000004      L.KRBA  000016      MC$NML  001274      MD$CI   000007
D$$L11= 000001          LP$TRB= 000002      L.LEN = 000022      MC$NMR  001142      MD$CNA  000003
D$$YNC= 000000          LP$UNT= 000001      L.MPF   000022      MC$NMS  001143      MD$COU  000001
D$$YNM= 000000          LP$WCN= 000040      L.NMST  000020      MC$NNO  001606      MD$DA   000010
E$$XPR= 000000          LP$WDV= 000020      L.NSTA  000014      MC$NNU  001605      MD$DL   000004
F$$LVL= 000001          LP$WLD= 000360      L.OWNR  000021      MC$NOP  001607      MD$DLV  000020
G$$TPP= 000000          LP$WTR= 000200      L.UNT   000013      MC$NOR  000000      MD$DMC  000014
G$$TSS= 000000          LP$WUN= 000100      MB$DIS  000001      MC$NPF  001616      MD$DMF  000046
G$$TTK= 000000          L$ADJ   000110      MB$ENA  000000      MC$NPR  001630      MD$DMP  000022
G$$WRD= 000000          L$BUF   000060      MB$FUL  000002      MC$NRE  001200      MD$DMR  000050
I$$RAR= 000000          L$CHN   000054      MB$MIX  000002      MC$NRT  001166      MD$DMV  000042
I$$RDN= 000000          L$COU   000114      MB$ONE  000001      MC$NVR  001642      MD$DN   000016
K$$CNT= 177546          L$CTB   000053      MB$RXO  000001      MC$P08  000000      MD$DP   000000
K$$CSR= 177546          L$CTL   000002      MB$TXO  000000      MC$P11  000001      MD$DPV  000044
K$$LDC= 000000          L$DDM   000000      MB$ZER  000000      MC$RFL  002046      MD$DQ   000006
K$$TPS= 000074          L$FLG   000064      MC$BAB  006073      MC$SBU  002051      MD$DTE  000024
LC$EXT  000000          L$FLX   000070      MC$BID  001765      MC$SFL  002044      MD$DU   000002
LC$INT  000001          L$LEN   000124      MC$BMC  001767      MC$SYC  000012      MD$DUP  000012
LD$LP = 000000          L$LTM   000070      MC$BSC  001766      MC$SYL  000036      MD$DV   000026
LF$END= 000020          L$MSG   000056      MC$CAC  001442      MC$SYR  000050      MD$DZ   000030
```

```
                                                                    E 15
CIRECO - READ/AND OR ZERO CIRCU MACRO V05.03b Saturday 29-Jun-85 12:17  Page 10-3
```

```
                                                .SBTTL  CIREIN - CIRCUIT READ INFORMATION INITIALIZE
                                        ;+
                                        ; **-CIREIN-CIRCUIT READ INFORMATION INITIALIZE ROUTINE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO INITIALIZE FOR THE CIRCUIT READ INFORMATION
                                        ; OPERATION.
                                        ;
                                        ; INPUTS:
                                        ;       R0 = OPTION (STATUS/SUMMARY/CHARACTERISTICS)
                                        ;       R1 = ID ADDRESS (LINE FORMAT)
                                        ;       R2 = LENGTH OF DATA IN INPUT BUFFER (ZERO)
                                        ;       R3 = ADDRESS OF INPUT BUFFER
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 = RETURN CODE
                                        ;       R2 = LENGTH OF DATA IN OUTPUT BUFFER
                                        ;       R3 = ADDESS OF OUTPUT BUFFER
                                        ;       DATA SAVED IN CONTEXT AREA
                                        ;       C-BIT = CLEAR IF SUCCESS, SET IF FAILURE
                                        ;
                                        ; REGISTERS:
                                        ;       R3, R4, R5 ARE PRESERVED
                                        ;-

 000000                                 CIREIN::
 000000  004567  000000G                        JSR     R5,$SAVRG               ; SAVE R3-R5
 000004  110064  000062                         MOVB    R0,L$OPT(R4)            ; SAVE OPTION BYTE
 000010  010364  000060                         MOV     R3,L$BUF(R4)            ; SAVE BUFFER POINTER
                                        ;
                                        ; CHECK LENGTH OF CONTEXT AREA
                                        ;
 000014  012700  177773                         MOV     #ME.MPR,R0              ; ASSUME CONTEXT AREA TOO SMALL
 000020  026427  177776  000124                 CMP     -2(R4),#L$LEN           ; IS CONTEXT AREA LARGE ENOUGH ?
 000026  002436                                 BLT     40$                     ; IF LT, NO - ERROR
                                        ;
                                        ; ZERO CONTEXT AREA
                                        ;
 000030  010400                                 MOV     R4,R0                   ; COPY CONTEXT POINTER
 000032  012705  000026                         MOV     #<L$CHN/2>,R5           ; SET NUMBER OF WORDS TO CLEAR
 000036  005020                         30$:    CLR     (R0)+                   ; CLEAR IT
 000040                                         SOB     R5,30$                  ; ...
                                        ;
                                        ; CHECK OPTIONS FLAGS
                                        ;
 000044  012700  177777                         MOV     #ME.FUN,R0              ; ASSUME ILLEGAL FUNCTION
 000050  132764  177717  000062                 BITB    #^C<MO$COU>,L$OPT(R4) ; ILLEGAL OPTIONS ?
 000056  001022                                 BNE     40$                     ; IF NE, YES - ERROR
                                          .IF DF R$$11S
                                                CMPB    #MO$CHA,L$OPT(R4)       ; NO CHARACTERISTICS FOR 11S
                                                BEQ     40$                     ; SO ERROR
                                          .ENDC
                                        ;
                                        ; MAKE SURE THERE IS NO INPUT DATA
                                        ;
 000060  012700  177776                         MOV     #ME.FOR,R0              ; ASSUME ILLEGAL INPUT FORMAT
 000064  005702                                 TST     R2                      ; ANY INPUT DATA ?
```

```
                                                               E  1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-5
Symbol table

S4.ANI= 000400      UC.NPR= 000100      U.ATT   000022      U.TST6  000056      W.BOFF  000012
S4.AVO= 001000      UC.PWF= 000020      U.BPKT= 000044      U.TTAB  000050      W.BPCB  000000
S4.BLK= 002000      UC.QUE= 000040      U.BUF   000024      U.TTYP  000046      W.BSIZ  000010
S4.DEC= 004000      UD.UNS= 000000      U.CBF = 000032      U.TUX   000024      XMTACT  001234R
S4.DLO= 000100      UD.160= 000004      U.CLI   177772      U.UHVR= 000107      X$$DBT= 000000
S4.EDT= 010000      UD.200= 000001      U.CNT   000030      U.UIC   000044      ZF.COU= 001000
S4.HFF= 000020      UD.556= 000002      U.COTQ  000030      U.UNFL= 000052      ZF.DDM= 000001
S4.HFL= 000007      UD.625= 000005      U.CTCB  000026      U.UNIT  000006      ZF.DIA= 004000
S4.HHT= 000040      UD.8K = 000006      U.CTL   000004      U.UNSZ= 000114      ZF.DLC= 000002
S4.HSY= 000200      UD.800= 000003      U.CTYP  000052      U.UNTI= 000060      ZF.DVP= 100000
S4.RGS= 020000      UM.CLI= 000036      U.CW1   000010      U.USVR= 000106      ZF.INI= 040000
S4.SFC= 040000      UM.DSB= 000200      U.CW2   000012      U.UTIL= 000036      ZF.KMX= 000020
S4.VFL= 000010      UM.NBR= 000400      U.CW3   000014      U.VCB = 000034      ZF.LLC= 000004
S5.ABP= 100000      UM.OVR= 000001      U.CW4   000016      U.VSER= 000120      ZF.LMC= 000100
S5.BCC= 020000      US.ABO= 000001      U.CYL = 000104      U.2MED= 000070      ZF.MAN= 020000
S5.DAO= 040000      US.BSP= 000002      U.DCB   000000      U2.AT.= 000020      ZF.MFL= 000010
S5.HPC= 000014      US.BSY= 000200      U.FCDE= 000042      U2.CRT= 002000      ZF.MTM= 000400
S5.HPO= 000020      US.FOR= 000040      U.FNUM= 000040      U2.DH1= 100000      ZF.MUX= 000040
S5.ITI= 000100      US.FRK= 000002      U.GRP = 000102      U2.DJ1= 040000      ZF.PSE= 002000
S5.OXF= 000040      US.KPF= 000001      U.KCSR= 000032      U2.DZ1= 000100      ZF.SLI= 010000
S5.RPO= 002000      US.LAB= 000004      U.KCS6= 000034      U2.ESC= 001000      ZF.TIM= 000200
S5.SW1= 000001      US.MDE= 000002      U.LUIC  177774      U2.HFF= 010000      ZF.X3P= 000000
S5.TMM= 000002      US.MDM= 000020      U.MEDI= 000040      U2.HLD= 000040      ZS.ASN= 100000
S5.VER= 010000      US.MNT= 000100      U.MLUN= 000050      U2.LOG= 000400      ZS.BSY= 140000
S5.XOF= 000004      US.OFL= 000001      U.MUP   177772      U2.LWC= 000001      Z.AVL   000014
S5.XON= 000010      US.PUB= 000004      U.OWN   177776      U2.L3S= 000004      Z.DAT   000016
S6.EIO= 000400      US.PWF= 000010      U.RBNS= 000112      U2.L8S= 010000      Z.DSP   000000
S6.RDI= 100000      US.RED= 000002      U.RCTC= 000113      U2.NEC= 004000      Z.FLG   000010
S6.RLU= 001000      US.SHR= 000001      U.RCTS= 000110      U2.PRV= 000010      Z.LEN = 000016
TADACT  000420R     US.SPU= 000002      U.RED   000002      U2.RMT= 020000      Z.LLN   000006
TH1ACT  001342R     US.UMD= 000010      U.RED2  000034      U2.R04= 100000      Z.MAP   000020
TH2ACT  001354R     US.VV = 000001      U.SCB   000020      U2.SCS= 000004      Z.NAM   000004
TH3ACT  001366R     US.WCK= 000010      U.SHST= 000076      U2.SLV= 000200      Z.PCB   000012
TR$BYT= 000004      UU.ABO= 000400      U.SHUN= 000074      U2.VT5= 000002      Z.SCH   000007
TR$EOM= 000010      UU.ATN= 000100      U.SPC = 000035      U2.7CH= 010000      $CAT5 = ****** GX
TR$ERR= 000016      UU.AVN= 000004      U.STS   000005      U3.OPA= 100000      $EXIT = 000000
TR$IMG= 000000      UU.GUS= 000014      U.ST2   000007      U3.PAR= 040000      $HEADR= ****** GX
TR$MAT= 000020      UU.IOS= 002000      U.SUB = 000036      U3.UPC= 020000      $SAVAL= ****** GX
TR$MBY= 000006      UU.ONL= 000020      U.TCHP  000042      U4.CR = 000100      $SAVRG= ****** GX
TR$NUL= 000012      UU.RCT= 000002      U.TCVP  000043      V$$CTR= 001000      $$$R  = 000010
TR$PAR= 000002      UU.RDY= 000200      U.TFLK  000040      W.BATT  000006      $$$SYS= 004374
TR$SUB= 000014      UU.SER= 000001      U.TFRQ  000036      W.BFPD  000014      $$$SYX= 000000
T$$KMG= 000000      UU.SIO= 001000      U.TIXL  000060      W.BHVR  000004      .MSGCN= ****** GX
T$$MIN= 000000      UU.SPC= 000040      U.TLPP  000053      W.BLGH  000020      .MSGNM= ****** GX
UC.ALG= 000200      U.ACB   000062      U.TMTI  000047      W.BLPD  000016      .MSGPT= ****** GX
UC.ATT= 000010      U.ACP = 000032      U.TRCK= 000100      W.BLVR  000002      .MSGST= ****** GX
UC.KIL= 000004      U.ADMA  000066      U.TSTA  000026      W.BNPD  000015      .NPARS= ****** GX
UC.LGH= 000003      U.AFLG  000064      U.TST5  000054

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        001412    001   (RW,I,LCL,REL,CON)
$MSTAT  002052    002   (RW,D,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics
```

```
                                                               F  1
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-6
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 12
MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME       REFERENCES

                  9-442     9-450     9-472
$PARAM           #8-82      9-226     9-239     9-246     9-254     9-262     9-270     9-277     9-293     9-300
                  9-307     9-316     9-323     9-330     9-337     9-344     9-351     9-358     9-365     9-372
                  9-379     9-386     9-393     9-400     9-407     9-414     9-423     9-430     9-437     9-447
                  9-455     9-462     9-469     9-477     9-492     9-497
$SBEXP           #8-82      8-88      8-90      8-94      8-97      8-100     8-103     8-106     8-109     8-112
                  8-115     8-118     8-121     8-124     8-127     8-130     8-133     8-136     8-139     8-142
                  8-145     8-148     8-151     8-154     8-157     8-160     8-163     8-166     8-169     8-172
                  8-175     8-178     8-181     8-184     8-187     8-194     8-197     8-200
$SKIP            #8-82      9-241     9-295     9-302
```

```
**FILE**ID**CICHOP
```

```
                                                .SBTTL  DSPTCH - DISPATCH TO PROPER ROUTINE

                                        ;+
                                        ;
                                        ; DSPTCH - DISPATCH TO PROPER SET OR CLEAR ROUTINE
                                        ;
                                        ; INPUTS:
                                        ;       R5          - POINTS TO DISPATCH TABLE
                                        ;       L$FL2(R4) - FLAGS WORD
                                        ;
                                        ; OUTPUTS:
                                        ;       THE PROPER ROUTINE IS CALLED TO PERFORM THE SET OR CLEAR FUNCTION
                                        ;
                                        ;-
000440  012700  000001                  DSPTCH: MOV     #MS.SUC,R0              ; ASSUME SUCCESS
000444  005715                                  TST     (R5)                    ; AT END OF TABLE ?
000446  001412                                  BEQ     30$                     ; IF EQ, YES - DONE
000450  032564  000070                          BIT     (R5)+,L$FL2(R4)         ; DISPATCH ON THIS OPERATION ?
000454  001403                                  BEQ     10$                     ; IF EQ, NO
000456                                          CALL    @(R5)+                  ; ELSE, DISPATCH TO ROUTINE
000460  103403                                  BCS     20$                     ; IF CS, ERROR - TRY TO CLEAN UP
000462  000766                                  BR      DSPTCH                  ; ELSE, LOOP FOR NEXT OPERATION
000464  005725                          10$:    TST     (R5)+                   ; SKIP OVER ROUTINE ADDRESS
000466  000764                                  BR      DSPTCH                  ; AND LOOK AT NEXT ENTRY
000470                                  20$:    .IF NDF R$$11S & S$$BAS & R$$RTR
000470                                          CALL    CLNUP                   ; TRY TO CLEAN UP AFTER AN ERROR
                                                .ENDC
000474                                  30$:    RETURN
```

```
                                                            E  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 4

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE            REFERENCES

LS.PRO  = 000002         #6-66
LS.PWF  = 040000         #6-66
LS.TOP  = 002000         #6-66
LS.UNF  = 020000         #6-66
LS.X25  = 000012         #6-66
LS.11D  = 004000         #6-66
LX.CEX  = 000004         #6-66
LX.LIN  = 000006         #6-66     12-362
LX.PRO  = 000005         #6-66
L$ACT     000104         #7-73
L$BBT     000114         #7-73
L$BLK     000262         #7-73
L$BSA     000122         #7-73
L$BSD     000123         #7-73
L$BSI     000124         #7-73
L$BUF     000060         #7-73    *9-153     9-232     9-256
L$CHN     000054         #7-73
L$CMB     000234         #7-73
L$COS     000101         #7-73
L$CTB     000053         #7-73
L$CTL     000002         #7-73
L$CUS     000260         #7-73
L$DDM     000000         #7-73
L$DDT     000106         #7-73
L$DEA     000105         #7-73
L$DLT     000110         #7-73
L$DTE     000240         #7-73
L$DTEL    000236         #7-73
L$DTEP    000250         #7-73
L$FLG     000064         #7-73     9-157    *9-221
L$FLX     000076         #7-73     9-165     9-209    *9-227
L$FL1     000066         #7-73     9-160     9-201    *9-223
L$FL2     000070         #7-73    *9-157     9-191    *9-201     10-280    11-322    12-359
L$HBT     000254         #7-73
L$HTM     000226         #7-73
L$INA     000125         #7-73
L$IND     000126         #7-73
L$INI     000127         #7-73
L$LCT     000224         #7-73
L$LEN     000264         #7-73
L$LMB     000234         #7-73
L$LTM     000230         #7-73
L$MRT     000253         #7-73
L$MSG     000056         #7-73    *9-155     9-242
L$MTYP    000075         #7-73
L$MWN     000252         #7-73
L$MXB     000133         #7-73
L$MXR     000253         #7-73
L$MXW     000252         #7-73
L$NAM     000000         #7-73
L$NLEN  = 000020         #7-73     7-73      7-73
L$NMT     000116         #7-73
```

```
                                                            F  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 5
```

```
                                        ; Circuits:
            000001                              LS$CHN =      1         ; Set PVC channel
            000002                              LS$CMB =      2         ; Set maximum data
            000004                              LS$CUS =      4         ; Set circuit usage
            000010                              LS$DTE =     10         ; Set DTE
            000020                              LS$MXR =     20         ; Set maximum recalls
            000040                              LS$MXW =     40         ; Set maximum window
            000100                              LS$NUM =    100         ; Set DLM number
            000200                              LS$RET =    200         ; Set recall timer
            001000                              LS$BLK =   1000         ; Set blocking
            100000                              LP$NXC = 100000         ; Circuit is a new X.25 circuit
                                        ; Lines:
            000400                              LS$HBT =    400         ; Set holdback timer
            000002                              LS$LMB = LS$CMB         ; Set max block
            000020                              LS$MRT = LS$MXR         ; Set max retransmits
            000040                              LS$MWN = LS$MXW         ; Set max window
            000200                              LS$NTI = LS$RET         ; Set retransmit timer
            010000                              LS$PRO =  10000         ; Set line protocol
                                        ; Common:
            020000                              LS$LCT =  20000         ; Set counter timer
                                        ; Groupings:
            020053                              LS$PVC = LS$CHN!LS$CMB!LS$DTE!LS$MXW!LS$LCT
            000362                              LS$SVC = LS$CMB!LS$MXR!LS$MXW!LS$NUM!LS$RET
            001004                              LS$DLM = LS$BLK!LS$CUS
```

```
                                         ;
001656  012700  017000          30$:     MOV     #<36*400>,R0            ; SET FUNCTION CODE
001662                                   CALL    DLXQIO                  ; ISSUE QIO
001666  012700  177747          40$:     MOV     #ME.OPE,R0              ; ASSUME OPERATION FAILED
001672  103403                           BCS     90$                     ; IF CS, IT FAILED
                                ;
                                ; SET CIRCUIT STATE TO ON, SUB-STATE TO CLOSED
                                ;
001674  112712  000000                   MOVB    #MS$ON,(R2)             ; SET NEW STATE
001700  000401                           BR      100$                    ; AND RETURN

001702  000261                  90$:     SEC
001704                          100$:    RETURN
```

```
                                        .SBTTL  DMP SET ROUTINES
                                ;+
                                ; DMP SET ROUTINES
                                ;
                                ; EACH ROUTINE POINTS TO THE PROPER TABLE AND CALLS DMPSET TO
                                ; SET THE PARAMETER.
                                ;
                                ;-
000222  012702  000120'         SETBBT::MOV     #BBTTBL,R2              ; POINT TO BABBLE TIMER TABLE
000226                                  CALL    DMPSET                  ; SET THE BABBLE TIMER
000232                                  RETURN
000234  012702  000132'         SETNMT::MOV     #NMTTBL,R2              ; POINT TO NORMAL TIMER TABLE
000240                                  CALL    DMPSET                  ; SET THE NORMAL TIMER
000244                                  RETURN
000246  012702  000144'         SETXMT::MOV     #XMTTBL,R2              ; POINT TO TRANSMIT DELAY TIMER TABLE
000252                                  CALL    DMPSET                  ; SET THE TRANSMIT DELAY TIMER
000256                                  RETURN
000260  012702  000156'         SETBSA::MOV     #BSATBL,R2              ; POINT TO ACTIVE BASE TABLE
000264                                  CALL    DMPSET                  ; SET THE ACTIVE BASE
000270                                  RETURN
000272  012702  000170'         SETBSD::MOV     #BSDTBL,R2              ; POINT TO DYING BASE TABLE
000276                                  CALL    DMPSET                  ; SET THE DYING BASE
000302                                  RETURN
000304  012702  000202'         SETBSI::MOV     #BSITBL,R2              ; POINT TO INACTIVE BASE TABLE
000310                                  CALL    DMPSET                  ; SET THE INACTIVE BASE
000314                                  RETURN
000316  012702  000214'         SETINA::MOV     #INATBL,R2              ; POINT TO ACTIVE INCREMENT TABLE
000322                                  CALL    DMPSET                  ; SET THE ACTIVE INCREMENT
000326                                  RETURN
000330  012702  000226'         SETIND::MOV     #INDTBL,R2              ; POINT TO DYING INCREMENT TABLE
000334                                  CALL    DMPSET                  ; SET THE DYING INCREMENT
000340                                  RETURN
000342  012702  000240'         SETINI::MOV     #INITBL,R2              ; POINT TO INACTIVE INCREMENT TABLE
000346                                  CALL    DMPSET                  ; SET THE INACTIVE INCREMENT
000352                                  RETURN
000354  012702  000252'         SETTH1::MOV     #TH1TBL,R2              ; POINT TO DEAD THRESHOLD TABLE
000360                                  CALL    DMPSET                  ; SET THE DEAD THRESHOLD
000364                                  RETURN
000366  012702  000264'         SETTH2::MOV     #TH2TBL,R2              ; POINT TO DYING THRESHOLD TABLE
000372                                  CALL    DMPSET                  ; SET THE DYING THRESHOLD
000376                                  RETURN
000400  012702  000276'         SETTH3::MOV     #TH3TBL,R2              ; POINT TO INACTIVE THRESHOLD TABLE
000404                                  CALL    DMPSET                  ; SET THE INACTIVE THRESHOLD
000410                                  RETURN
000412  012702  000310'         SETMXB::MOV     #MXBTBL,R2              ; POINT TO MAXIMUM BLOCKS TABLE
000416                                  CALL    DMPSET                  ; SET THE MAXIMUM BLOCKS
000422                                  RETURN
```

```
                                                                  E  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 4
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE             REFERENCES

LS$RET  = 000200           #6-86      6-86      6-86
LS$SER  = 100000           #6-86
LS$STA  = 000002           #6-86
LS$SVC  = 000362           #6-86
LS$TAD  = 000020           #6-86
LS$TH1  = 000100           #6-86
LS$TH2  = 000200           #6-86
LS$TH3  = 000400           #6-86
LS$XMT  = 040000           #6-86
LS.AST  = 100000           #5-71
LS.CEX  = 000001           #5-71
LS.CIR  = 000010           #5-71
LS.CXO  = 010000           #5-71
LS.CX5  = 000400           #5-71
LS.DLM  = 004000           #5-71
LS.ECH  = 001000           #5-71
LS.FDX  = 004000           #5-71
LS.HDX  = 010000           #5-71
LS.LIN  = 000003           #5-71
LS.LMC  = 000007           #5-71
LS.NTI  = 004000           #5-71
LS.ON   = 000011           #5-71
LS.OPT  = 000400           #5-71      9-169
LS.PRO  = 000002           #5-71
LS.PWF  = 040000           #5-71
LS.TOP  = 002000           #5-71
LS.UNF  = 020000           #5-71
LS.X25  = 000012           #5-71
LS.11D  = 004000           #5-71
LX.CEX  = 000004           #5-71
LX.LIN  = 000006           #5-71
LX.PRO  = 000005           #5-71
L$ACT     000104           #6-86
L$BBT     000114           #6-86      8-130
L$BLK     000262           #6-86
L$BSA     000122           #6-86      8-133
L$BSD     000123           #6-86      8-134
L$BSI     000124           #6-86      8-135
L$BUF     000060           #6-86
L$CHN     000054           #6-86      18-745    19-796    20-891
L$CMB     000234           #6-86
L$COS     000101           #6-86      20-876    20-882
L$CTB     000053           #6-86
L$CTL     000002           #6-86      11-229
L$CUS     000260           #6-86
L$DDM     000000           #6-86      11-228    15-598    24-1122
L$DDT     000106           #6-86
L$DEA     000105           #6-86
L$DLT     000110           #6-86
L$DTE     000240           #6-86
L$DTEL    000236           #6-86
L$DTEP    000250           #6-86
```

```
                                                                  F  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 5
```

```
        000020                          LP$WDV = 20               ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40               ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100              ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200              ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK

                                  ;
                                  ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                  ;
        000001                          LS$NTL=1                  ; NTL SET FUNCTION
        000002                          LS$STA=2                  ; SET STATE
        000004                          LS$COS=4                  ; SET COST
        000010                          LS$OWN=10                 ; SET OWNER
        000020                          LS$TAD=20                 ; SET TRIBUTARY ADDRESS
        000040                          LS$ACT=40                 ; SET MULTIPOINT ACTIVE
        000100                          LS$DEA=100                ; SET MULTIPOINT DEAD
        000200                          LC$NTL=200                ; NTL CLEAR FUNCTION
        000400                          LC$OWN=400                ; CLEAR OWNER
        001000                          LS$DDT=1000               ; SET DEAD TIMER
        002000                          LS$DLT=2000               ; SET DELAY TIMER
        004000                          LS$PLT=4000               ; SET POLL TIMER
        010000                          LS$BBT=10000              ; SET BABBLE TIMER
        020000                          LS$NMT=20000              ; SET NORMAL TIMER
        040000                          LS$XMT=40000              ; SET TRANSMIT TIMER
        100000                          LS$SER=100000             ; SET SERVICE [ENABLE/DISABLE]

                                  ;
                                  ; FLAGS WORD BIT DEFINITIONS (L$FL1)
                                  ;
        000001                          LS$BSA=1                  ; SET ACTIVE BASE
        000002                          LS$BSD=2                  ; SET DYING BASE
        000004                          LS$BSI=4                  ; SET INACTIVE BASE
        000010                          LS$INA=10                 ; SET ACTIVE INCREMENT
        000020                          LS$IND=20                 ; SET DYING INCREMENT
        000040                          LS$INI=40                 ; SET INACTIVE INCREMENT
        000100                          LS$TH1=100                ; SET DEAD THRESHOLD
        000200                          LS$TH2=200                ; SET DYING THRESHOLD
        000400                          LS$TH3=400                ; SET INACTIVE THRESHOLD
        001000                          LS$MXB=1000               ; SET MAXIMUM BLOCKS
        010000                          LS$HTM=10000              ; SET HELLO TIMER
        020000                          LS$MDT=20000              ; SET MODEM TEST
        040000                          LS$LOO=40000              ; SET CONTROLLER LOOPBACK
        100000                          LS$NOR=100000             ; SET CONTROLLER NORMAL
                                        .IF DF  R$$PRO  ; PRO/DECNET
                                  ;
                                  ; LOOPBACK TYPE WORD BIT DEFINITIONS (L$LTY)
                                  ;
                                        LS$INT = 0                ; INTERNAL LOOPBACK
                                        LS$EXT = 1                ; EXTERNAL LOOPBACK
                                  ;
                                  ; MODEM TEST TYPE WORD BIT DEFINITIONS (L$MDT)
                                  ;
                                        LS$LCL = 0                ; LOCAL MODEM TEST
                                        LS$REM = 1                ; REMOTE MODEM TEST
                                        LS$OFF = 2                ; SHUT OFF MODEM TESTS
                                        .ENDC   ; DF R$$PRO
                                  ;
                                  ; Flags word bit definitions for X.25 circuits and lines (L$FLX)
                                  ;
```

```
                                                                     E 10
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 16-2
Symbol table

MB$ONE  000001     MC$NVR  001642     MD$DN   000016     ME$MOD  000004     ML$CLS  000000
MB$RXO  000001     MC$P08  000000     MD$DP   000000     ME$NOD  000000     ML$CON  000001
MB$TXO  000000     MC$P11  000001     MD$DPV  000044     ME$NON  177777     ML$EXT  000001
MB$ZER  000000     MC$RFL  002046     MD$DQ   000006     ME$OBJ  000007     ML$FIL  000002
MC$BAB  006073     MC$SBU  002051     MD$DTE  000024     ME$OB2  000004     ML$FIR  000000
MC$BID  001765     MC$SFL  002044     MD$DU   000002     ME$PRO  000005     ML$INT  000000
MC$BMC  001767     MC$SYC  000012     MD$DUP  000012     ME$SYS  000006     ML$KNO  140000
MC$BSC  001766     MC$SYL  000036     MD$DV   000026     ME.BLO  177744     ML$MON  000003
MC$CAC  001442     MC$SYR  000050     MD$DZ   000030     ME.CON  177753     ML$SYS  000002
MC$CAP  001440     MC$SYS  000024     MD$FUL  000000     ME.CST  177765     ML$TOP  000001
MC$CCL  001445     MC$SYZ  000000     MD$HAL  000001     ME.DIS  177755     MN$UNL  000377
MC$CDC  002045     MC$TBR  001140     MD$HEL  000002     ME.DON  177600     MO$ACC  000200
MC$CDP  001441     MC$TBS  001141     MD$KCP  000013     ME.FCO  177762     MO$ADD  000002
MC$CIF  001465     MC$TDR  002053     MD$KDP  000034     ME.FIO  177756     MO$ALA  000004
MC$CLD  001464     MC$TIM  000000     MD$KDZ  000036     ME.FOP  177763     MO$ALI  000000
MC$COU  100000     MC$TYP  007777     MD$KL   000040     ME.FOR  177776     MO$CHA  000040
MC$CTL  001454     MC$UBU  002052     MD$KMX  000054     ME.FUN  177777     MO$CIR  000003
MC$CTR  001452     MC$UFD  002047     MD$KMY  000052     ME.GRO  177745     MO$CLE  000100
MC$CTS  001453     MC$UMR  001132     MD$LEN  000050     ME.HAR  177750     MO$COU  000060
MC$DOV  002050     MC$UMS  001133     MD$PCL  000011     ME.IID  177767     MO$DAC  000000
MC$LBR  001750     MC$VAX  000003     MD$QNA  000005     ME.LCO  177766     MO$DEF  000000
MC$LBS  001751     MC$WID  060000     MD$UNA  000001     ME.LPR  177757     MO$DPR  000000
MC$LDI  001774     MC$WIH  040000     MD$WIT  000002     ME.MPR  177773     MO$ENT  000017
MC$LDO  001775     MC$WIL  020000     MD.ABO  000017     ME.MVE  177771     MO$EVE  000100
MC$LDR  001762     MC$W08  020000     MD.ACC  000010     ME.OPE  177747     MO$INF  000160
MC$LDS  001763     MC$W16  040000     MD.AOB  000016     ME.PLO  177751     MO$INS  000002
MC$LLB  002021     MC$W32  060000     MD.BOB  000011     ME.PMI  177743     MO$LIN  000001
MC$LLP  002115     MC$XBR  001750     MD.DIA  000007     ME.PNA  177752     MO$LOG  000001
MC$LLR  002007     MC$XBS  001751     MD.DOB  000015     ME.PRI  177775     MO$MIR  000031
MC$LOO  000001     MC$XCJ  000322     MD.DUM  000002     ME.PTY  177772     MO$NAM  000001
MC$LRB  002020     MC$XCR  002260     MD.FAI  000014     ME.PVA  177760     MO$NIC  000023
MC$LRP  002114     MC$XCS  002261     MD.FNA  000001     ME.RES  177761     MO$NOD  000000
MC$LRR  002006     MC$XDR  001762     MD.FOB  000006     ME.ROO  177754     MO$OFF  000001
MC$LSI  002032     MC$XDS  001763     MD.LOA  000001     ME.SIZ  177774     MO$ON   000000
MC$LST  002033     MC$XFR  002272     MD.LSH  000020     ME.SYS  177746     MO$OPT  000001
MC$LUP  006072     MC$XFS  002273     MD.NOB  000012     ME.UCO  177770     MO$PER  000200
MC$MAP  010000     MC$XLJ  000323     MD.NON  177777     MF$ACT  177776     MO$PRO  000002
MC$MBL  001764     MC$XLR  002330     MD.PER  000000     MF$ADD  000000     MO$PRS  000001
MC$MBX  004420     MC$XMA  000310     MD.RES  000004     MF$ADJ  177774     MO$REA  000200
MC$MBY  001752     MC$XMC  002305     MD.ROB  000005     MF$ALL  177775     MO$SET  000000
MC$MPX  004406     MC$XMS  002304     MD.RSH  000013     MF$BYE  000300     MO$STA  000020
MC$NAP  001604     MC$XNR  002332     MD.SDU  000005     MF$CHA  000023     MO$SUM  000000
MC$NBR  001130     MC$XRC  002316     MD.SLO  000003     MF$DUM  000020     MO$TSK  000000
MC$NBS  001131     MC$XRR  002331     MD.SNA  000000     MF$EVT  000001     MO$VOL  000000
MC$NCR  001154     MC$XRS  002342     MD.TLO  000004     MF$FUP  000264     MO$ZER  000000
MC$NCS  001155     MC$020  000002     MD.UNA  000002     MF$KNO  177777     MP$ACB  000012
MC$NML  001274     MD$CI   000007     MD.UNR  000003     MF$LOA  000017     MP$ACC  000514
MC$NMR  001142     MD$CNA  000003     MD.UOB  000007     MF$LOO  177775     MP$ACT  004526
MC$NMS  001143     MD$COU  000001     MD.VOL  000006     MF$REA  000024     MP$ADD  000766
MC$NNO  001606     MD$DA   000010     ME$ALI  000010     MF$SIG  177773     MP$ADJ  001440
MC$NNU  001605     MD$DL   000004     ME$AL2  000003     MF$SPF  000302     MP$ADP  006325
MC$NOP  001607     MD$DLV  000020     ME$ARE  000005     MF$SYS  000026     MP$ADS  004406
MC$NOR  000000     MD$DMC  000014     ME$CIR  000003     MF$TES  000022     MP$ALB  000036
MC$NPF  001616     MD$DMF  000046     ME$EXA  000000     MF$TRA  000301     MP$AMC  001640
MC$NPR  001630     MD$DMP  000022     ME$EXE  000200     MF$TRI  000021     MP$AMH  001641
MC$NRE  001200     MD$DMR  000050     ME$LIN  000001     MF$ZER  000025     MP$ANB  004420
MC$NRT  001166     MD$DMV  000042     ME$LOG  000002     ML$ALL  100000     MP$ASB  000024
```

```
                                                                     F 10
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 16-3
```

```
        .TITLE  CIRECH - READ CIRCUIT CHARACTERISTICS
        .IDENT  /V05.00/
        .ENABL  LC
        .NLIST  CND
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - READ CIRCUIT CHARACTERISTICS INFORMATION
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  01-JUL-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
 457                                                .SBTTL  FNDHTM - Find hello timer
 458
 459                                        ;+
 460                                        ;
 461                                        ; FNDHLT - FIND HELLO TIMER
 462                                        ;
 463                                        ; INPUTS:
 464                                        ;       R4 - ADDRESS OF CONTEXT AREA
 465                                        ;
 466                                        ; OUTPUTS:
 467                                        ;       CARRY CLEAR:
 468                                        ;           HTIM - HELLO TIMER VALUE
 469                                        ;
 470                                        ;-
 471 001402                                 FNDHTM:
 472 001402                                         SWSTK$  30$                             ;; ENTER SYSTEM STATE
 473 001406                                         SAVMAP                                  ;; SAVE PREVIOUS MAPPING
 474 001412                                         CALL    FNDCTR                          ;; LOOK FOR COUNTER BLOCK
 475 001416 103006                                  BCC     10$                             ;; BR IF FOUND IT
 476 001420                                         RETC    R0                              ;; ELSE SET USER C-BIT
 477 001432 000415                                  BR      20$                             ;; AND EXIT
 478 001434 016167 000030 176350    10$:    MOV     T$T5(R1),HTIM                   ;; SAVE HELLO TIMER
 479 001442 012767 000001 176346            MOV     #MB$DIS,BLKING                  ;; Assume no DLM blocking       ;[TD001]
 480 001450 132761 000040 000044            BITB    #PX$BLK,T$FLAG(R1)              ;; Check blocking flag          ;[TD001]
 481 001456 001403                                  BEQ     20$                             ;; Branch if not set            ;[TD001]
 482 001460 012767 000000 176330            MOV     #MB$ENA,BLKING                  ;; Indicate blocking enabled    ;[TD001]
 483 001466                         20$:    RESMAP                                  ;; RESTORE PREVIOUS MAPPING
 484 001472                         30$:    RETURN                                  ;; EXIT SYSTEM STATE
```

```
                                                                         E 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-3
Symbol table

MO$ALI   000000      MP$BUF   000170      MP$GRP   000541      MP$MVE   000145      MP$RRT   002213
MO$CHA   000040      MP$BUP   006324      MP$GTY   002224      MP$MVI   001634      MP$RST   002212
MO$CIR   000003      MP$BUS   001643      MP$HAD   001757      MP$MVR   001751      MP$RSV   001754
MO$CLE   000100      MP$CAC   001750      MP$HBT   002142      MP$MWN   002154      MP$RSZ   001756
MO$COU   000060      MP$CAS   001762      MP$HDD   000162      MP$MXB   002172      MP$RTI   001616
MO$DAC   000000      MP$CAT   002210      MP$HTM   001612      MP$MXC   000466      MP$RTM   001611
MO$DEF   000000      MP$CCS   004406      MP$HWA   002210      MP$MXR   001630      MP$RVE   001604
MO$DPR   000000      MP$CHN   002141      MP$IAT   001322      MP$MXW   002143      MP$RVT   000156
MO$ENT   000017      MP$CIR   000144      MP$IDE   000144      MP$NAA   006335      MP$SAC   000514
MO$EVE   000100      MP$CLK   002131      MP$IDP   006327      MP$NAC   001130      MP$SAD   000543
MO$INF   000160      MP$CLN   002126      MP$IHO   000215      MP$NAP   006326      MP$SCA   000310
MO$INS   000002      MP$CLT   002211      MP$INA   002177      MP$NCT   000240      MP$SCO   000144
MO$LIN   000001      MP$CMB   002142      MP$IND   002204      MP$NET   002114      MP$SCT   002176
MO$LOG   000001      MP$CMK   000537      MP$INI   002201      MP$NLI   000765      MP$SDU   000203
MO$MIR   000031      MP$CMX   002153      MP$ITI   000776      MP$NNA   000764      MP$SDV   000160
MO$NAM   000001      MP$CND   000310      MP$LAA   000231      MP$NOD   000500      MP$SEH   004401
MO$NIC   000023      MP$CNU   001753      MP$LAN   000234      MP$NRA   006334      MP$SER   000144
MO$NOD   000000      MP$COB   000311      MP$LAR   000202      MP$NRB   000214      MP$SET   000000
MO$OFF   000001      MP$CON   002126      MP$LAS   000012      MP$NSA   001617      MP$SGZ   001644
MO$ON    000000      MP$COS   001604      MP$LBS   000203      MP$NTI   002141      MP$SID   000176
MO$OPT   000001      MP$CPF   000230      MP$LCO   000226      MP$NUM   001642      MP$SIN   000310
MO$PER   000200      MP$CPL   000232      MP$LCT   000156      MP$NVE   001274      MP$SLI   000156
MO$PRO   000002      MP$CPT   000226      MP$LHL   000232      MP$NXN=  001476      MP$SLO   000171
MO$PRS   000001      MP$CPU   000161      MP$LLE   000227      MP$OAC   000632      MP$SMX   002202
MO$REA   000200      MP$CSZ   001755      MP$LLO   004432      MP$OAN   000620      MP$SND   000500
MO$SET   000000      MP$CUS   002127      MP$LMB   002152      MP$OCO   000776      MP$SNP   006332
MO$STA   000020      MP$CVA   000540      MP$LMX   002200      MP$OHO   000214      MP$SNU   000542
MO$SUM   000000      MP$DAL   002570      MP$LNA   000144      MP$ONA   000764      MP$SOB   000524
MO$TSK   000000      MP$DCO=  001464      MP$LNO   000233      MP$ONR   002114      MP$SPA   000157
MO$VOL   000000      MP$DDT   002177      MP$LOA   000170      MP$OQL   001453      MP$SPR   000536
MO$ZER   000000      MP$DEL   001131      MP$LOG   004514      MP$OTI   000777      MP$SPS   000513
MP$ACB   000012      MP$DES   000156      MP$LOO   000620      MP$OTY   001022      MP$STA   000000
MP$ACC   000514      MP$DEV   002114      MP$LTM   001613      MP$OUS   000777      MP$STI   002140
MP$ACT   004526      MP$DFA   001320      MP$LTY   004533      MP$OVE   001010      MP$STT   002201
MP$ADD   000766      MP$DHO=  001465      MP$LWI   000230      MP$OWN   004374      MP$STY   000175
MP$ADJ   001440      MP$DIA   000173      MP$MAC   004420      MP$PAR   000036      MP$SUB   000001
MP$ADP   006325      MP$DLB   002571      MP$MAD   001630      MP$PAS   000513      MP$SUR   000156
MP$ADS   004406      MP$DLG   004521      MP$MAP   006323      MP$PCH   002152      MP$SUS   000512
MP$ALB   000036      MP$DLI=  001466      MP$MAR   001635      MP$PCO   000024      MP$SVR   000163
MP$AMC   001640      MP$DLT   002200      MP$MAV   006321      MP$PCT   000144      MP$SWI   002260
MP$AMH   001641      MP$DRO   001441      MP$MBN   001636      MP$PDT   002114      MP$SYP   002424
MP$ANB   004420      MP$DST   000454      MP$MBR   001637      MP$PHA   000170      MP$TFL   000156
MP$ASB   000024      MP$DTE   002140      MP$MBU   001642      MP$PHY   000012      MP$TH1   002206
MP$ASC   004432      MP$DTY=  001452      MP$MCB   000156      MP$PLI   000025      MP$TH2   002205
MP$AUS   000512      MP$DUA   000207      MP$MCO   001632      MP$PLN   002140      MP$TH3   002202
MP$BBT   002165      MP$DUC   000210      MP$MDE   004421      MP$PLO   000012      MP$TLN   000202
MP$BDF   002164      MP$DUM   000202      MP$MHO   001633      MP$PMC   004374      MP$TLO   000172
MP$BFQ   002121      MP$DUP   002127      MP$MLB   000202      MP$PNT   002126      MP$TPA   004375
MP$BLK   001616      MP$DVC   002114      MP$MLK   001306      MP$PRI   004411      MP$TRI   002164
MP$BLO   001452      MP$DWE   001321      MP$MLN   001631      MP$PRO   002130      MP$TST   000144
MP$BMX   002176      MP$ELT   000157      MP$MLP   006333      MP$PSS   001763      MP$TYP   002130
MP$BNP   006330      MP$ETY   001605      MP$MRB   002171      MP$PST   001762      MP$UCS   004407
MP$BRT   001620      MP$EVE   000311      MP$MRO   001605      MP$RET   001631      MP$USR   001750
MP$BSA   002176      MP$FNC   001752      MP$MRP   006322      MP$RFA   001323      MP$VEC   004410
MP$BSD   002203      MP$GDT   002222      MP$MRT   002153      MP$RMX   002201      MP$VER   004406
MP$BSI   002200      MP$GNM   002223      MP$MRV   006320      MP$R[illegible]  004374      MP$WDF   002165
MP$BSP   006331      MP$GRO   002115      MP$MSB   000170      MP$RP[illegible]  001606      MP$WMX   002177
```

```
                                                                         F 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-4
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 10

MACRO CROSS REFERENCE                                    CREF   04.00

MACRO NAME      REFERENCES

.LSTHD          #6-74       6-74
.LTAB           #6-74
.MPLHD          #6-74       6-74
.MXPTB          #6-74
.PECHA          #6-74
.POOL           #6-74
.PRI            #6-74       6-74       6-74
.SCOM           #6-74
.SECSR          #6-74
.SLNB           #6-74       6-74
.SLNW           #6-74       6-74
.STNB           #6-74       6-74
.STNW           #6-74
.TIME           #6-74
.UNB            #6-74       6-74
.UNW            #6-74
.VFY            #6-74
.X2CHB          #6-74
.X2CHW          #6-74
.X3CHB          #6-74
.X3CHW          #6-74
```

```
**FILE**ID**CIRECO
```

```
Symbol table

MD$FUL  000000          ME.CST  177765          ML$TOP  000001          MP$BRT  001620          MP$EVE  000311
MD$HAL  000001          ME.DIS  177755          MN$UNL  000377          MP$BSA  002176          MP$FNC  001752
MD$HEL  000002          ME.DON  177600          MO$ACC  000200          MP$BSD  002203          MP$GDT  002222
MD$KCP  000013          ME.FCO  177762          MO$ADD  000002          MP$BSI  002200          MP$GNM  002223
MD$KDP  000034          ME.FIO  177756          MO$ALA  000004          MP$BSP  006331          MP$GRD  002115
MD$KDZ  000036          ME.FOP  177763          MO$ALI  000000          MP$BUF  000170          MP$GRP  000541
MD$KL   000040          ME.FOR  177776          MO$CHA  000040          MP$BUP  006324          MP$GTY  002224
MD$KMX  000054          ME.FUN  177777          MO$CIR  000003          MP$BUS  001643          MP$HAD  001757
MD$KMY  000052          ME.GRO  177745          MO$CLE  000100          MP$CAC  001750          MP$HBT  002142
MD$LEN  000050          ME.HAR  177750          MO$COU  000060          MP$CAS  001762          MP$HDD  000162
MD$PCL  000011          ME.IID  177767          MO$DAC  000000          MP$CAT  002210          MP$HTM  001612
MD$QNA  000005          ME.LCO  177766          MO$DEF  000000          MP$CCS  004406          MP$HWA  002210
MD$UNA  000001          ME.LPR  177757          MO$DPR  000000          MP$CHN  002141          MP$IAT  001322
MD$WIT  000002          ME.MPR  177773          MO$ENT  000017          MP$CIR  000144          MP$IDE  000144
MD.ABO  000017          ME.MVE  177771          MO$EVE  000100          MP$CLK  002131          MP$IDP  006327
MD.ACC  000010          ME.OPE  177747          MO$INF  000160          MP$CLN  002126          MP$IHO  000215
MD.AOB  000016          ME.PLO  177751          MO$INS  000002          MP$CLT  002211          MP$INA  002177
MD.BOB  000011          ME.PMI  177743          MO$LIN  000001          MP$CMB  002142          MP$IND  002204
MD.DIA  000007          ME.PNA  177752          MO$LOG  000001          MP$CMK  000537          MP$INI  002201
MD.DOB  000015          ME.PRI  177775          MO$MIR  000031          MP$CMX  002153          MP$ITI  000776
MD.DUM  000002          ME.PTY  177772          MO$NAM  000001          MP$CND  000310          MP$LAA  000231
MD.FAI  000014          ME.PVA  177760          MO$NIC  000023          MP$CNU  001753          MP$LAN  000234
MD.FNA  000001          ME.RES  177761          MO$NOD  000000          MP$COB  000311          MP$LAR  000202
MD.FOB  000006          ME.ROO  177754          MO$OFF  000001          MP$CON  002126          MP$LAS  000012
MD.LOA  000001          ME.SIZ  177774          MO$ON   000000          MP$COS  001604          MP$LBS  000203
MD.LSH  000020          ME.SYS  177746          MO$OPT  000001          MP$CPF  000230          MP$LCO  000226
MD.NOB  000012          ME.UCO  177770          MO$PER  000200          MP$CPL  000232          MP$LCT  000156
MD.NON  177777          MF$ACT  177776          MO$PRO  000002          MP$CPT  000226          MP$LHL  000232
MD.PER  000000          MF$ADD  000000          MO$PRS  000001          MP$CPU  000161          MP$LLE  000227
MD.RES  000004          MF$ADJ  177774          MO$REA  000200          MP$CSZ  001755          MP$LLO  004432
MD.ROB  000005          MF$ALL  177775          MO$SET  000000          MP$CUS  002127          MP$LMB  002152
MD.RSH  000013          MF$BYE  000300          MO$STA  000020          MP$CVA  000540          MP$LMX  002200
MD.SDU  000005          MF$CHA  000023          MO$SUM  000000          MP$DAL  002570          MP$LNA  000144
MD.SLO  000003          MF$DUM  000020          MO$TSK  000000          MP$DCO= 001464          MP$LNO  000233
MD.SNA  000000          MF$EVT  000001          MO$VOL  000000          MP$DDT  002177          MP$LOA  000170
MD.TLO  000004          MF$FUP  000264          MO$ZER  000000          MP$DEL  001131          MP$LOG  004514
MD.UNA  000002          MF$KNO  177777          MP$ACB  000012          MP$DES  000156          MP$LOO  000620
MD.UNR  000003          MF$LOA  000017          MP$ACC  000514          MP$DEV  002114          MP$LTM  001613
MD.UOB  000007          MF$LOO  177775          MP$ACT  004526          MP$DFA  001320          MP$LTY  004533
MD.VOL  000006          MF$REA  000024          MP$ADD  000766          MP$DHO= 001465          MP$LWI  000230
ME$ALI  000010          MF$SIG  177773          MP$ADJ  001440          MP$DIA  000173          MP$MAC  004420
ME$AL2  000003          MF$SPF  000302          MP$ADP  006325          MP$DLB  002571          MP$MAD  001630
ME$ARE  000005          MF$SYS  000026          MP$ADS  004406          MP$DLG  004521          MP$MAP  006323
ME$CIR  000003          MF$TES  000022          MP$ALB  000036          MP$DLI= 001466          MP$MAR  001635
ME$EXA  000000          MF$TRA  000301          MP$AMC  001640          MP$DLT  002200          MP$MAV  006321
ME$EXE  000200          MF$TRI  000021          MP$AMH  001641          MP$DRO  001441          MP$MBN  001636
ME$LIN  000001          MF$ZER  000025          MP$ANB  004420          MP$DST  000454          MP$MBR  001637
ME$LOG  000002          ML$ALL  100000          MP$ASB  000024          MP$DTE  002140          MP$MBU  001642
ME$MOD  000004          ML$CLS  000000          MP$ASC  004432          MP$DTY= 001452          MP$MCB  000156
ME$NOD  000000          ML$CON  000001          MP$AUS  000512          MP$DUA  000207          MP$MCD  001632
ME$NON  177777          ML$EXT  000001          MP$BBT  002165          MP$DUC  000210          MP$MDE  004421
ME$OBJ  000007          ML$FIL  000002          MP$BDF  002164          MP$DUM  000202          MP$MHO  001633
ME$OB2  000004          ML$FIR  000000          MP$BFQ  002121          MP$DUP  002127          MP$MLB  000207
ME$PRO  000005          ML$INT  000000          MP$BLK  001616          MP$DVC  002114          MP$MLK  001306
ME$SYS  000006          ML$KND  140000          MP$BLD  001452          MP$DWE  001321          MP$MLN  001631
ME.BLO  177744          ML$MON  000003          MP$BMX  002176          MP$ELT  000157          MP$MLP  006333
ME.CON  177753          ML$SYS  000002          MP$BNP  006330          MP$ETY  001605          MP$MRB  002171
```

```
CIREIN - CIRCUIT READ INFORMATION INITIALIZE

 000066  001016                          BNE     40$                     ; IF NE, YES - ILLEGAL
                                 ;
                                 ; CHECK AND SAVE LINE FORMAT
                                 ;
 000070  121127  177773                  CMPB    (R1),#MF$SIG            ; SHOW SIGNIFIGANT CIRCUITS ?
 000074  001003                          BNE     32$                     ; NO
 000076  052764  000040  000064          BIS     #LF$SIG,L$FLG(R4)       ; ELSE SET FLAG BIT

 000104  012764  000003  000120  32$:    MOV     #ME$CIR,L$PAR(R4)       ; SET ERROR DETAIL
 000112                                  CALL    CHKCIR                  ; CHECK LINE FORMAT
 000116  103402                          BCS     40$                     ; IF CS, ERROR
 000120  012700  000002                  MOV     #MS.MOR,R0              ; ELSE, MORE TO COME
                                 ;
                                 ; USE THE STATUS IN R0 TO BUILD THE RETURN MESSAGE AND C-BIT RETURN
                                 ;
 000124                          40$:
                                         .IF     DF,R$$MPL

                                         SWSTK$  42$
                                         SAVMAP                          ;; SAVE CURRENT KISAR6 MAPPING
                                         MOV     @SAHDB,@KSAR6           ;; MAP TO MY EXTERNAL HEADER
                                         MOV     @SAHPT,R3               ;; POINT TO THE HEADER
                                         MOV     H.LUN+<<$MBLUN-1>*4>(R3),R$R3+2(SP) ; POINT TO NS: UCB IN USER R3
                                         RESMAP                          ;; RESTORE PREVIOUS KISAR6
                                         RETURN                          ;; EXIT TO USER STATE
                                 42$:
                                         .IFF    ; DF R$$MPL

 000124  017703  000000G                 MOV     @HEADR,R3               ; POINT TO OUR HEADER           ; RJK02
 000130  016303  000000C                 MOV     H.LUN+<<$MBLUN-1>*4>(R3),R3 ; POINT TO NS: UCB       ;**-1

                                         .ENDC   ; DF R$$MPL

 000134  032763  000200  000014          BIT     #NF.END,U.CW3(R3)       ; IS EXECUTOR AN ENDNODE ?
 000142  001403                          BEQ     45$                     ; IF EQ, NO
 000144  052764  000020  000064          BIS     #LF$END,L$FLG(R4)       ; ELSE REMEMBER IT
 000152  016403  000060          45$:    MOV     L$BUF(R4),R3            ; RESTORE THE BUFFER ADDRESS
 000156  052764  000001  000064          BIS     #LF$REA,L$FLG(R4)       ; SAY OPERATION IS READ INFORMATION
 000164  012702  000001                  MOV     #1,R2                   ; SET MESSAGE LENGTH TO 1 BYTE
 000170  110013                          MOVB    R0,(R3)                 ; SET THE RETURN CODE
 000172  100013                          BPL     60$                     ; IF PL, SUCCESS
 000174  116463  000120  000001          MOVB    L$PAR(R4),1(R3)         ; STORE LOW BYTE OF ERROR DETAIL
 000202  116463  000121  000002          MOVB    L$PAR+1(R4),2(R3)       ; STORE HIGH BYTE
 000210  012702  000003                  MOV     #3,R2                   ; UPDATE LENGTH OF MESSAGE
 000214  005064  000064          50$:    CLR     L$FLG(R4)               ; ELSE, UNSUCCESSFUL INIT
 000220  000261                          SEC                             ; ELSE, NO - SET ERROR RETURN
 000222                          60$:    RETURN

         000001                          .END
```

F 1

```
CICHIN - CIRCUIT CHANGE PARAMET MACRO V05.03b Saturday 29-Jun-85 12:10  Page 42-6
Symbol table

Work  file  reads: 299
Work  file writes: 208
Size of work file: 37463 Words  ( 147 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:58.05
SY:CICHIN.V2,[135,134]CICHIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CICHIN
```

G 1

```
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 1
```

```
                                                                             F  2
**FILE**ID**CICHOP




  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     OOOOOO     PPPPPPPP
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     OOOOOO     PPPPPPPP
CC               II       CC           HH      HH   OO      OO   PP      PP
CC               II       CC           HH      HH   OO      OO   PP      PP
CC               II       CC           HH      HH   OO      OO   PP      PP
CC               II       CC           HH      HH   OO      OO   PP      PP
CC               II       CC           HHHHHHHHHH   OO      OO   PPPPPPPP
CC               II       CC           HHHHHHHHHH   OO      OO   PPPPPPPP
CC               II       CC           HH      HH   OO      OO   PP
CC               II       CC           HH      HH   OO      OO   PP
CC               II       CC           HH      HH   OO      OO   PP               ....
CC               II       CC           HH      HH   OO      OO   PP               ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     OOOOOO     PP               ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH     OOOOOO     PP               ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                                                             G  2
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12
Table of contents
```

```
                                        .SBTTL  SETUP - SET UP FOR COMMAND PROCESSING
                                ;+
                                ; **-SETUP-SET UP FOR COMMAND PROCESSING
                                ;
                                ; THIS ROUTINE IS CALLED TO INITALIZE THE CURRENT OPERATION FLAGS
                                ; FOR A CIRCUIT.
                                ;
                                ; INPUTS:
                                ;       R4 = ADDRESS OF CONTEXT BLOCK WITH:
                                ;               L$FL2(R4) = FLAGS FOR THIS CIRCUIT
                                ;
                                ; OUTPUTS:
                                ;       IF A STATE CHANGE WAS REQUESTED, AND THE LINE IS NOT
                                ;       LOADED, AN ERROR MESSAGE IS RETURNED
                                ;
                                ; REGISTERS:
                                ;       R4, R5 ARE PRESERVED
                                ;
                                ;-

000476                          SETUP:
                                        .IF NDF R$$11S
000476  022764  177777  000050          CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
000504  001421                          BEQ     30$                     ; BR IF YES
                                        .IFTF   ; NDF R$$11S
000506  016400  000042                  MOV     L$SLT(R4),R0            ; GET SLT ADDRESS
                                        .IFT    ; NDF R$$11S
                                ;
                                ; IF A STATE CHANGE IS REQUESTED, AND THE CIRCUIT IS NOT SET, MARK
                                ; AN NTL SET LINE REQUEST
                                ;
000512  032764  000002  000070          BIT     #LS$STA,L$FL2(R4)       ; STATE CHANGE REQUESTED ?
000520  001413                          BEQ     30$                     ; IF EQ, NO
                                        .ENDC   ; NDF R$$11S
000522  032760  040000  000000          BIT     #LF.RDY,L.FLG(R0)       ; IS THE LINE LOADED ?
000530  001007                          BNE     30$                     ; IF NE, YES - PROCEED
000532  012700  177765                  MOV     #ME.CST,R0              ; INDICATE COMPONENT IN WRONG STATE
000536  012764  000001  000072          MOV     #ME$LIN,L$PAR(R4)       ; COMPONENT = LINE
000544  000261                          SEC                             ; INDICATE ERROR
000546  000400                          BR      30$                     ; AND RETURN
000550                          30$:    RETURN

                                        .IF NDF R$$11S & S$$BAS & R$$RTR
```

```
                                                                  F  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 5

SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE           REFERENCES

L$NTI     000256        #7-73
L$NTL     000134        #7-73       12-361    *12-362    *12-364
L$NUM     000240        #7-73
L$NUML    000236        #7-73
L$NXT     000044        #7-73
L$OPT     000062        #7-73
L$OWN     000102        #7-73
L$PAR     000072        #7-73      *9-154     *9-235      9-236      9-237     *11-328
L$PDV     000055        #7-73
L$PFG     000041        #7-73       9-252
L$PLT     000112        #7-73
L$PRO     000075        #7-73
L$PVC     000004        #7-73
L$RET     000256        #7-73
L$SCN     000020        #7-73
L$SCR     000166        #7-73
L$SER     000230        #7-73
L$SLEN    000074        #7-73
L$SLT     000042        #7-73       9-178      11-316     13-391
L$SNM     000046        #7-73
L$STA     000100        #7-73
L$TAD     000103        #7-73
L$TH1     000130        #7-73
L$TH2     000131        #7-73
L$TH3     000132        #7-73
L$TNM     000052        #7-73
L$TPT     000050        #7-73       9-176      11-313     13-395
L$TRB     000040        #7-73       13-407
L$TYP     000063        #7-73
L$UNT     000003        #7-73
L$XCH     000232        #7-73
L$XMT     000120        #7-73
L.COST    000015        #6-68
L.CTL     000012        #6-68
L.CVA     177776        #6-68
L.DDM     000002        #6-68
L.DDS     000004        #6-68
L.DLC     000003        #6-68
L.DLM     000006        #6-68
L.DLS     000010        #6-68
L.FLG     000000        #6-68       11-325
L.KRBA    000016        #6-68
L.LEN   = 000022        #6-68
L.MPF     000022        #6-68
L.NMST    000020        #6-68
L.NSTA    000014        #6-68       13-393
L.OWNR    000021        #6-68       13-392
L.UNT     000013        #6-68
ME$LIN    000001         11-328
ME.CST    177765         11-327
ME.DON    177600         9-156      9-219
MS.SUC    000001         10-277
```

```
                                                                  G  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 6
```

```
     88                                      .SBTTL  ERROR MESSAGE STRINGS
     89
     90
     91                                      .IF NDF R$$PRO  ; PRO/DECNET
     92
     93                              ;
     94                              ; ERROR MESSAGE STRINGS
     95                              ;
     96                                      .ENABL  LC
     97                                      .IF NDF R$$11S
     98 000000                       INSMSG: MESAG$  <Network Loader not installed>
     99 000035                       NTLMSG: MESAG$  <Network Loader function failed>
    100                                      .ENDC
    101 000074                       NODMSG: MESAG$  <Node in wrong state>
    102
    103                                      .ENDC   ; NDF R$$PRO
    104
    105                                      .EVEN
    106
    107                                      .IF NDF S$$BAS & R$$RTR & R$$11S
    108
    109                                      .IF NDF R$$PRO  ; PRO/DECNET
    110
```

```
                                        .SBTTL  STASER - REQUEST CIRCUIT STATE "SERVICE"
                                ;+
                                ; **-STASER-REQUEST CIRCUIT STATE TO "SERVICE"
                                ;
                                ; THIS ROUTINE IS CALLED TO CHECK THE CURRENT CIRCUIT STATE, AND
                                ; IF LEGAL, CHANGE IT TO SERVICE.
                                ;
                                ; INPUTS:
                                ;       R2 = ADDRESS OF STATE/OWNER WORD IN SLT
                                ;       R4 = ADDRESS OF CONTEXT BLOCK
                                ;
                                ; OUTPUTS:
                                ;       IF CC, SUCCESS
                                ;       ELSE, R0 CONTAINS ERROR CODE
                                ;-

001706                          STASER:
                                ;
                                ; CHECK FOR ILLEGAL STATES
                                ;
001706  032762  000400  177760          BIT     #LF.BRO,L.FLG-L.NMST(R2) ; IS THIS A BROADCAST CHANNEL
001714  001004                          BNE     3$                      ; IF NE, N.G.
001716  026427  000000  015355          CMP     L$DDM(R4),#^RDLM        ; ELSE, DLM CIRCUIT ?
001724  001006                          BNE     5$                      ; IF NE, NO - OK
001726  012700  177760          3$:     MOV     #ME.PVA,R0              ; ELSE PARAMETER VALUE ERROR
001732  012764  000000  000072          MOV     #MP$STA,L$PAR(R4)       ; ... STATE
001740  000455                          BR      90$                     ; RETURN ERROR

001742  011201                  5$:     MOV     (R2),R1                 ; GET CURRENT STATE AND OWNER
001744  010100                          MOV     R1,R0                   ; ...
001746  142701  177417                  BICB    #^C360,R1               ; ISOLATE SUB-STATE BITS
001752  140100                          BICB    R1,R0                   ; ISOLATE STATE BITS
001754  120027  000002                  CMPB    R0,#MS$SER              ; IS THE CIRCUIT ALREADY IN SERVICE ?
001760  001446                          BEQ     100$                    ; IF EQ, YES - NO-OP
001762  120027  000001                  CMPB    R0,#MS$OFF              ; ELSE, IS STATE OFF ?
001766  001416                          BEQ     20$                     ; IF EQ, YES - SKIP SUB-STATE CHECK
001770  120027  000000                  CMPB    R0,#MS$ON               ; ELSE, IS IT ON ?
001774  001005                          BNE     10$                     ; IF NE, NO - ERROR
001776  105701                          TSTB    R1                      ; IS SUB-STATE CLOSED ?
002000  001411                          BEQ     20$                     ; IF EQ, YES - LEGAL
002002  120127  000040                  CMPB    R1,#<LN.REF*20>         ; IS THE SUB-STATE REFLECTING ?
002006  001424                          BEQ     40$                     ; IF EQ, YES - LEGAL, BUT SKIP DLX QIOS
002010  012700  177765          10$:    MOV     #ME.CST,R0              ; ELSE, CIRCUIT IN WRONG STATE
002014  012764  000003  000072          MOV     #ME$CIR,L$PAR(R4)       ; ...
002022  000424                          BR      90$                     ; AND RETURN WITH ERROR
                                ;
                                ; REASSIGN CIRCUIT TO DLX
                                ;
002024  012700  017020          20$:    MOV     #<36*400!20>,R0         ; SET REASSIGN FUNCTION CODE
002030  012701  015370                  MOV     #^RDLX,R1               ; SET NEW LLC NAME (DLX)
002034                                  CALL    DLXQIO                  ; ISSUE QIO
002040  103404                          BCS     30$                     ; IF CS, OPERATION FAILED
                                ;
                                ; TELL DLX THAT THE CIRCUIT IS ON
                                ;
002042  012700  017000                  MOV     #<36*400>,R0            ; SET ON FUNCTION CODE
002046                                  CALL    DLXQIO                  ; ISSUE QIO
```

```
                                        .SBTTL  DMPSET - DMP SET PARAMETERS ROUTINE

                                ;+
                                ; **-DMPSET-SET THE DMP PARAMETERS
                                ;
                                ; INPUTS:
                                ;       R2 - POINTER TO DMP PARAMETER TABLE
                                ;       R4 - POINTER TO CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       CARRY CLEAR:
                                ;                    THE PARAMETER IS SET IN THE LINE TABLE
                                ;       CARRY SET:
                                ;                    SET OPERATION FAILED, R0 = ERROR CODE
                                ;
                                ;       R4,R5 PRESERVED
                                ;
                                ;-
000424  012700  177752          DMPSET: MOV     #ME.PNA,R0              ; ASSUME PARAMETER NOT APPLICABLE
000430  012264  000072                  MOV     (R2)+,L$PAR(R4)         ; SET PARAMETER VALUE
000434  022764  177777  000050          CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
000442  001447                          BEQ     30$                     ; BR IF YES
000444  012703  177777                  MOV     #-1,R3                  ; ASSUME APPLIED ACROSS ALL TRIBS
000450  005722                          TST     (R2)+                   ; APPLY PARAMETER ACROSS ALL TRIBS?
000452  001413                          BEQ     10$                     ; BR IF YES
000454  116403  000040                  MOVB    L$TRB(R4),R3            ; GET TRIB OF CURRENT CIRCUIT
000460  132764  000002  000041          BITB    #LP$TRB,L$PFG(R4)       ; WAS A TRIB SPECIFIED?
000466  001015                          BNE     20$                     ; BR IF YES
000470  132764  000200  000041          BITB    #LP$WTR,L$PFG(R4)       ; WILD CARD TRIB SPECIFIED?
000476  001431                          BEQ     30$                     ; BR IF NO, ERROR
000500  000410                          BR      20$                     ; YES, ATTEMPT TO SET THE PARAMETER
000502  132764  000002  000041  10$:    BITB    #LP$TRB,L$PFG(R4)       ; WAS A TRIB SPECIFIED?
000510  001404                          BEQ     20$                     ; BR IF NO, O.K.
000512  132764  000200  000041          BITB    #LP$WTR,L$PFG(R4)       ; WILD CARD TRIB SPECIFIED?
000520  001420                          BEQ     30$                     ; BR IF NO, ERROR
000522  132764  000010  000041  20$:    BITB    #LP$MPT,L$PFG(R4)       ; IS THE CIRCUIT A MULTIPOINT CIRCUIT?
000530  001414                          BEQ     30$                     ; BR IF NO, ERROR
000532  005064  000072                  CLR     L$PAR(R4)               ; CLEAR ERROR DETAIL
000536  010401                          MOV     R4,R1                   ; POINT TO CONTEXT AREA
000540  062201                          ADD     (R2)+,R1                ; POINT TO SAVED PARAMETER VALUE
000542  005000                          CLR     R0                      ; STORE PARAMETER VALUE IN R0
000544  156100  000001                  BISB    1(R1),R0                ; STORE PARAMATER VALUE HIGH BYTE
000550  000300                          SWAB    R0                      ; STORE IN HIGH BYTE
000552  151100                          BISB    (R1),R0                 ; STORE LOW BYTE
000554                                  CALL    $DMPST                  ; FIND DMP LINE TABLE
000560  000401                          BR      40$                     ; FINISHED
000562  000261                  30$:    SEC                             ; INDICATE ERROR
000564                          40$:    RETURN

                                        .ENDC   ; NDF S$$BAS & R$$RTR & R$$PRO
```

```
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE           REFERENCES

L$FLG     000064          #6-86
L$FLX     000076          #6-86
L$FL1     000066          #6-86
L$FL2     000070          #6-86
L$HBT     000254          #6-86
L$HTM     000226          #6-86       12-303
L$INA     000125          #6-86       8-136
L$IND     000126          #6-86       8-137
L$INI     000127          #6-86       8-138
L$LCT     000224          #6-86
L$LEN     000264          #6-86
L$LMB     000234          #6-86
L$LTM     000230          #6-86
L$MRT     000253          #6-86
L$MSG     000056          #6-86      *11-244     *11-254     *13-457
L$MTYP    000075          #6-86
L$MWN     000252          #6-86
L$MXB     000133          #6-86       8-142
L$MXR     000253          #6-86
L$MXW     000252          #6-86
L$NAM     000000          #6-86
L$NLEN  = 000020          #6-86       6-86        6-86
L$NMT     000116          #6-86       8-131
L$NTI     000256          #6-86
L$NTL     000134          #6-86       9-169      *11-228     *11-229     *11-230      11-233
L$NUM     000240          #6-86
L$NUML    000236          #6-86
L$NXT     000044          #6-86
L$OPT     000062          #6-86
L$OWN     000102          #6-86       21-989      21-1000     21-1002
L$PAR     000072          #6-86      *9-175      *12-285     *12-290     *12-294     *12-308     *13-339     *13-347     *13-414
                         *14-546     *15-601     *15-619     *17-694     *18-727     *20-858     *20-862     *20-872     *21-959
                         *23-1077    *23-1095    *24-1144
L$PDV     000055          #6-86       14-552      18-744      19-795      20-892
L$PFG     000041          #6-86       9-165       10-199      10-201      13-368      20-863      20-865      21-960      21-962
                           23-1084    23-1086     23-1089     23-1091     23-1093
L$PLT     000112          #6-86
L$PRO     000075          #6-86
L$PVC     000004          #6-86
L$RET     000256          #6-86
L$SCN     000020          #6-86
L$SCR     000166          #6-86       11-234
L$SER     000230          #6-86       24-1131     24-1138
L$SLEN    000074          #6-86
L$SLT     000042          #6-86       9-171       12-288      13-345      18-728      18-758      19-809      20-867      21-973
                           24-1127
L$SNM     000046          #6-86
L$STA     000100          #6-86       13-351      13-360      13-374      13-419      13-481      13-488      13-492      17-696
                           17-700
L$TAD     000103          #6-86
L$TH1     000130          #6-86       8-139
L$TH2     000131          #6-86       8-140
```

```
                                        ; Circuits:
             000001                             LS$CHN =       1        ; Set PVC channel
             000002                             LS$CMB =       2        ; Set maximum data
             000004                             LS$CUS =       4        ; Set circuit usage
             000010                             LS$DTE =      10        ; Set DTE
             000020                             LS$MXR =      20        ; Set maximum recalls
             000040                             LS$MXW =      40        ; Set maximum window
             000100                             LS$NUM =     100        ; Set DLM number
             000200                             LS$RET =     200        ; Set recall timer
             001000                             LS$BLK =    1000        ; Set blocking
             100000                             LP$NXC = 100000         ; Circuit is a new X.25 circuit
                                        ; Lines:
             000400                             LS$HBT =     400        ; Set holdback timer
             000002                             LS$LMB = LS$CMB         ; Set max block
             000020                             LS$MRT = LS$MXR         ; Set max retransmits
             000040                             LS$MWN = LS$MXW         ; Set max window
             000200                             LS$NTI = LS$RET         ; Set retransmit timer
             010000                             LS$PRO =   10000        ; Set line protocol
                                        ; Common:
             020000                             LS$LCT =   20000        ; Set counter timer
                                        ; Groupings:
             020053                             LS$PVC = LS$CHN!LS$CMB!LS$DTE!LS$MXW!LS$LCT
             000362                             LS$SVC = LS$CMB!LS$MXR!LS$MXW!LS$NUM!LS$RET
             001004                             LS$DLM = LS$BLK!LS$CUS
    105
```

```
Symbol table

MP$ASC  004432      MP$DTY= 001452      MP$MCB  000156      MP$PLI  000025      MP$TH2  002205
MP$AUS  000512      MP$DUA  000207      MP$MCO  001632      MP$PLN  002140      MP$TH3  002202
MP$BBT  002165      MP$DUC  000210      MP$MDE  004421      MP$PLO  000012      MP$TLN  000202
MP$BDF  002164      MP$DUM  000202      MP$MHO  001633      MP$PMC  004374      MP$TLO  000172
MP$BFQ  002121      MP$DUP  002127      MP$MLB  000202      MP$PNT  002126      MP$TPA  004375
MP$BLK  001616      MP$DVC  002114      MP$MLK  001306      MP$PRI  004411      MP$TRI  002164
MP$BLO  001452      MP$DWE  001321      MP$MLN  001631      MP$PRO  002130      MP$TST  000144
MP$BMX  002176      MP$ELT  000157      MP$MLP  006333      MP$PSS  001763      MP$TYP  002130
MP$BNP  006330      MP$ETY  001605      MP$MRB  002171      MP$PST  001762      MP$UCS  004407
MP$BRT  001620      MP$EVE  000311      MP$MRO  001605      MP$RET  001631      MP$USR  001750
MP$BSA  002176      MP$FNC  001752      MP$MRP  006322      MP$RFA  001323      MP$VEC  004410
MP$BSD  002203      MP$GDT  002222      MP$MRT  002153      MP$RMX  002201      MP$VER  004406
MP$BSI  002200      MP$GNM  002223      MP$MRV  006320      MP$RPA  004374      MP$WDF  002165
MP$BSP  006331      MP$GRO  002115      MP$MSB  000170      MP$RPR  001606      MP$WMX  002177
MP$BUF  000170      MP$GRP  000541      MP$MVE  000145      MP$RRT  002213      MP$XMT  002166
MP$BUP  006324      MP$GTY  002224      MP$MVI  001634      MP$RST  002212      MP$XPF  000242
MP$BUS  001643      MP$HAD  001757      MP$MVR  001751      MP$RSV  001754      MP$XPL  000244
MP$CAC  001750      MP$HBT  002142      MP$MWN  002154      MP$RSZ  001756      MP$XPT  000240
MP$CAS  001762      MP$HDD  000162      MP$MXB  002172      MP$RTI  001616      MP$XXX  177777
MP$CAT  002210      MP$HTM  001612      MP$MXC  000466      MP$RTM  001611      MP$5ST  004406
MP$CCS  004406      MP$HWA  002210      MP$MXR  001630      MP$RVE  001604      MS$ACT  000001
MP$CHN  002141      MP$IAT  001322      MP$MXW  002143      MP$RVT  000156      MS$ADU  000010
MP$CIR  000144      MP$IDE  000144      MP$NAA  006335      MP$SAC  000514      MS$ALO  000007
MP$CLK  002131      MP$IDP  006327      MP$NAC  001130      MP$SAD  000543      MS$ASE  000006
MP$CLN  002126      MP$IHO  000215      MP$NAP  006326      MP$SCA  000310      MS$ATR  000011
MP$CLT  002211      MP$INA  002177      MP$NCT  000240      MP$SCO  000144      MS$AUT  000000
MP$CMB  002142      MP$IND  002204      MP$NET  002114      MP$SCT  002176      MS$CLE  000003
MP$CMK  000537      MP$INI  002201      MP$NLI  000765      MP$SDU  000203      MS$DED  000004
MP$CMX  002153      MP$ITI  000776      MP$NNA  000764      MP$SDV  000160      MS$DIE  000003
MP$CND  000310      MP$LAA  000231      MP$NOD  000500      MP$SEH  004401      MS$DUM  000004
MP$CNU  001753      MP$LAN  000234      MP$NRA  006334      MP$SER  000144      MS$FAI  000013
MP$COB  000311      MP$LAR  000202      MP$NRB  000214      MP$SET  000000      MS$GLO  000000
MP$CON  002126      MP$LAS  000012      MP$NSA  001617      MP$SGZ  001644      MS$HOL  000002
MP$COS  001604      MP$LBS  000203      MP$NTI  002141      MP$SID  000176      MS$INA  000002
MP$CPF  000230      MP$LCO  000226      MP$NUM  001642      MP$SIN  000310      MS$LOA  000003
MP$CPL  000232      MP$LCT  000156      MP$NVE  001274      MP$SLI  000156      MS$LOO  000002
MP$CPT  000226      MP$LHL  000232      MP$NXN= 001476      MP$SLO  000171      MS$OFF  000001
MP$CPU  000161      MP$LLE  000227      MP$OAC  000632      MP$SMX  002202      MS$ON   000000
MP$CSZ  001755      MP$LLO  004432      MP$OAN  000620      MP$SND  000500      MS$PRO  000006
MP$CUS  002127      MP$LMB  002152      MP$OCO  000776      MP$SNP  006332      MS$REA  000004
MP$CVA  000540      MP$LMX  002200      MP$OHO  000214      MP$SNU  000542      MS$REF  000001
MP$DAL  002570      MP$LNA  000144      MP$ONA  000764      MP$SOB  000524      MS$RES  000003
MP$DCO= 001464      MP$LNO  000233      MP$ONR  002114      MP$SPA  000157      MS$RST  000001
MP$DDT  002177      MP$LOA  000170      MP$OQL  001453      MP$SPR  000536      MS$RSX  000002
MP$DEL  001131      MP$LOG  004514      MP$OTI  000777      MP$SPS  000513      MS$RT   000005
MP$DES  000156      MP$LOO  000620      MP$OTY  001022      MP$STA  000000      MS$SER  000002
MP$DEV  002114      MP$LTM  001613      MP$OUS  000777      MP$STI  002140      MS$SHU  000002
MP$DFA  001320      MP$LTY  004533      MP$OVE  001010      MP$STT  002201      MS$SRV  000007
MP$DHO= 001465      MP$LWI  000230      MP$OWN  004374      MP$STY  000175      MS$STA  000000
MP$DIA  000173      MP$MAC  004420      MP$PAR  000036      MP$SUB  000001      MS$SYN  000012
MP$DLB  002571      MP$MAD  001630      MP$PAS  000513      MP$SUR  000156      MS$TER  000001
MP$DLG  004521      MP$MAP  006323      MP$PCH  002152      MP$SUS  000512      MS$TOP  000003
MP$DLI= 001466      MP$MAR  001635      MP$PCO  000024      MP$SVR  000163      MS$TRI  000005
MP$DLT  002200      MP$MAV  006321      MP$PCT  000144      MP$SWI  002260      MS$UNR  000005
MP$DRO  001441      MP$MBN  001636      MP$PDT  002114      MP$SYP  002424      MS$VMS  000004
MP$DST  000454      MP$MBR  001637      MP$PHA  000170      MP$TFL  000156      MS.DON  177600
MP$DTE  002140      MP$MBU  001642      MP$PHY  000012      MP$TH1  002206      MS.MOR  000002
```

```
     55                                       .SBTTL  Macro calls
     56                               ;
     57                               ; MACRO LIBRARY CALLS
     58                               ;
     59                                       .MCALL  MANDF$,SAVRG,RESRG,PDVDF$,SLTDF$,PCLDF$,MESAG$,PHBDF$
     60                                       .MCALL  DDCDF$,SAVMAP,MAP,RESMAP,RETC,CEACC$,ECDDB$
     61                                       .MCALL  CTRDF$,BIAS,SVCDF$,XPDDB$,PLBDF$,CIRCX$,ASL$,ADJDF$,DHBDF$
     62
     63 000000                                CTRDF$                          ; DEFINE TRANSPORT COUNTER BLOCK SYMBOLS
     64 000000                                ADJDF$                          ; DEFINE ADJACENCY OFFSETS
     65 000000                                DHBDF$                          ; DEFINE DEC HOME BLOCK OFFSETS
     66 000000                                MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     67 000000                                ECDDB$                          ; DEFINE ECL DDB OFFSETS
     68 000000                                XPDDB$                          ; DEFINE XPT DDB OFFSETS
     69 000000                                PDVDF$                          ; DEFINE PDV OFFSETS
     70 000000                                SLTDF$                          ; DEFINE SLT OFFSETS
     71 000000                                SVCDF$                          ; DEFINE SVC OFFSETS
     72 000000                                DDCDF$                          ; DEFINE DDCMP STATION TABLE OFFSETS
     73 000000                                PHBDF$                          ; DEFINE PSI HOME BLOCK OFFSETS
     74 000000                                PCLDF$                          ; DEFINE PCL LINE TABLE OFFSETS
     75 000000                                PLBDF$                          ; DEFINE PLB OFFSETS
     76                               ;
     77                               ; LOCAL DEFINITIONS
     78                               ;
     79                000002                 RBIAS=2                         ; WE MUST ACCOUNT FOR SAVED KISAR6
     80
     81 000000                                CIRCX$  LIST,<$X>
                                      ;
                                      ; CONTEXT AREA OFFSET DEFINITIONS
                                      ;
        000210                                .ASECT
                       000000                 . = 0
                       000020                 L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                      ;
                                      ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                      ; as in the CIRCX$ and LOCCX$ macros.
                                      ;
        000000                        L$NAM:
        000000                        L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                        L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                        L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                        L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                       000020                 . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                        L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                        L$TRB:  .BLKB   1                       ; TRIBUTARY NUMBER
        000041                        L$PFG:  .BLKB   1                       ; PARSE FLAGS
        000042                        L$SLT:  .BLKW   1                       ; Current SLT/PVC address
        000044                        L$NXT:  .BLKW   1                       ; Pointer to next SLT/PVC address
        000046                        L$SNM:  .BLKW   1                       ; REMAINING NUMBER OF SYSTEM LINES
        000050                        L$TPT:  .BLKW   1                       ; Current tributary pointer (-1 for PSI)
        000052                        L$TNM:  .BLKB   1                       ; REMAINING NUMBER OF TRIBUTARIES
        000053                        L$CTB:  .BLKB   1                       ; CURRENT TRIBUTARY NUMBER
        000054                        L$CHN:  .BLKB   1                       ; Channel number / X.25 port number and
        000055                        L$PDV:  .BLKB   1                       ; ... PDV assigned
        000056                        L$MSG:  .BLKW   1                       ; ERROR MESSAGE STRING POINTER
        000060                        L$BUF:  .BLKW   1                       ; SAVED BUFFER POINTER
        000062                        L$OPT:  .BLKB   1                       ; SAVED OPTIONS BYTE
```

```
                                                  .SBTTL  FNDLTM - Find adjacent node and listen timer
                                        ;+
                                        ;** FNDLTM - Find adjacent node and listen timer
                                        ;
                                        ; Inputs:       L$FLG(R4):  LF$SEG if partial response in progress
                                        ;               L$BUF(R4) = address of response message
                                        ;
                                        ; Output:       Carry set:   no information.
                                        ;               Carry clear: success -
                                        ;
                                        ;               LTIM = current listen timer
                                        ;               L$FLG(R4):  LF$SEG set if partial response message
                                        ;              @L$BUF(R4) = MS.PAR if partial response message
                                        ;               L$LTM(R4) = encoded listener timer
                                        ;               L$NOD(R4) = node address
                                        ;               L$ADJ(R4) = virtual address of next adjacency
                                        ;               L$COU(R4) = number of unscanned adjacencies
                                        ;
                                        ; Registers:    No registers modified
                                        ;-

 001474                                 FNDLTM: SAVRG   <R1,R2>                         ; save registers
 001500                                         CALL    FNDPLD                          ; look for PLB for channel
 001504 103002                                  BCC     1$                              ; if CC, circuit owned by transport
 001506 000167  000432                          JMP     90$                             ; if CS, no info

 001512                                 1$:     SWSTK$  60$                             ;; enter system state
 001516 010205                                  MOV     R2,R5                           ;; copy PLB address
 001520 012702  114224                          MOV     #^RXPT,R2                       ;; get process id
 001524                                         CALL    @PDVID                          ;; get PDV index                ; RJK10
 001530 103551                                  BCS     30$                             ;; if CS, get out now
 001532 120264  000055                          CMPB    R2,L$PDV(R4)                    ;; is circuit owned by XPT ?
 001536 001146                                  BNE     30$                             ;; if NE, no
 001540 067702  000000G                         ADD     @PDVTA,R2                       ;; point into PDV vector        ; RJK10
 001544 011200                                  MOV     (R2),R0                         ;; point to process PDV
 001546 005760  000000                          TST     Z.DSP(R0)                       ;; is process loaded ?
 001552 001540                                  BEQ     30$                             ;; if EQ, no
 001554 016002  000016                          MOV     Z.DAT(R0),R2                    ;; else point to process DDB

 001560 032764  000020  000064                  BIT     #LF$END,L$FLG(R4)               ;; is executor an endnode ?
 001566 001432                                  BEQ     5$                              ;; if EQ, no
                                        ;
                                        ; All circuits (Endnode): return listen timer for single adjacency
                                        ;
 001570                                         MAP     Z.DSP(R0)                       ;; map to XPT process
 001576 016201  000072                          MOV     N$ADJ1(R2),R1                   ;; get ADJ1 virtual address
 001602                                         BIAS    R1                              ;; bias it
 001612 132761  000010  000002                  BITB    #AT$UP,A$TYP(R1)                ;; is adjacency up ?
 001620 001515                                  BEQ     30$                             ;; if EQ, no information
 001622 016167  000000  176150                  MOV     A$NID(R1),NODE                  ;; get adjacent node address
 001630 016202  000074                          MOV     N$ADJ2(R2),R2                   ;; get ADJ2 virtual address
 001634                                         BIAS    R2                              ;; bias it for APR6
 001644 116266  000002  000000G                 MOVB    A$TMI(R2),R$R1(SP)              ;; return listen timer
 001652 000513                                  BR      55$                             ;; return success

 001654 132765  000200  000001  5$:     BITB    #PT$BRO,P$TYP(R5)               ; is circuit a broadcast channel ?
 001662 001027                                  BNE     10$                             ; if NE, yes
```

```
Symbol table

MP$XMT  002166      MT$LPB  000005      NF$DMO= 000010      N$VCB   000010      PX$SVC= 000100
MP$XPF  000242      MT$MAX  000037      NF$MOU= 000040      N$VER   000066      P$CHN   000004
MP$XPL  000244      MT$MUL  000100      NF$RST= 000002      N$XLEN  000124      P$CNT   000005
MP$XPT  000240      MT$NLE  000017      NF$SCN= 000020      N$$ACC= 000001      P$CTR   000034
MP$XXX  177777      MT$NON  000001      NF$SHU= 000004      N$$BUF= 000001      P$DRCT  000015
MP$5ST  004406      MT$NR4  000005      NF$TIM= 000200      N$$LDV= 000001      P$DRTR  000036
MS$ACT  000001      MT$NTY  000060      NM.CLN  003400      N$$MCP= 000001      P$FLG   000012
MS$ADU  000010      MT$OCT  000060      NM.INI  002400      N$$MLL= 000001      P$FWD   000030
MS$ALO  000007      MT$PHA  000002      NM.DPR  003000      N$$MOV= 000010      P$ICCB  000046
MS$ASE  000006      MT$POI  000000      NM.VR2  002401      N$$NCT= 000001      P$IPL   000014
MS$ATR  000011      MT$QP2  000010      NM.VR3  002402      N$$PEM= 000001      P$LCD   000002
MS$AUT  000000      MT$ROU  000000      NDDE    000000R     OWNER   000002R     P$LEN   000052
MS$CLE  000003      MT$RO4  000004      N$ACQ   000000      PDVID = ****** GX   P$LST   000000
MS$DED  000004      MT$SEC  000000      N$ACTL  000032      PDVTA = ****** GX   P$NRNI  000040
MS$DIE  000003      MT$SGD  000020      N$ADJ1  000072      PF$BLK= 020000      P$OCCB  000050
MS$DUM  000004      MT$SYS  000002      N$ADJ2  000074      PF$CLC= 010000      P$PFQ   000006
MS$FAI  000013      MT$TER  000001      N$CACH  000062      PF$DLM= 100000      P$PKSZ  000044
MS$GLO  000000      MT$TRI  000002      N$CIR   000034      PF$EIP= 000002      P$RMX1  000016
MS$HOL  000002      MT$TYP  007777      N$CRC   000120      PF$ENB= 000001      P$RMX2  000020
MS$INA  000002      MT$USD  000000      N$DLA   000020      PF$FAI= 004000      P$RPRI  000042
MS$LOA  000003      MT$X25  000003      N$DLY   000014      PF$FM1= 000100      P$RTIM  000003
MS$LOO  000002      MU$INC  000001      N$ELEN  000054      PF$FM2= 000200      P$STA1  000022
MS$OFF  000001      MU$OUT  000002      N$ENC   000042      PF$OFF= 000000      P$STA2  000023
MS$ON   000000      MU$PER  000000      N$ERRC  000022      PF$RM1= 000020      P$TIM   000010
MS$PRO  000006      MV$III  000000      N$FLG   000005      PF$RM2= 000040      P$TSCT  000026
MS$REA  000004      MV$IV   000001      N$FNC   000006      PF$RVR= 000010      P$TSIZ  000024
MS$REF  000001      MX$ACT  000020      N$GENQ  000052      PF$STA= 000004      P$TYP   000001
MS$RES  000003      MX$CIR  000020      N$GTM   000015      PF$SVC= 040000      P$$P45= 000000
MS$RST  000001      MX$CLN  000020      N$HC1   000052      PF$UP = 000004      P$$WRD= 000000
MS$RSX  000002      MX$CNM  000006      N$HC2   000056      PR$BED= 000200      Q$$OPT= 000010
MS$RT   000005      MX$CON  000006      N$HIGH  000033      PR$BEU= 000100      RBIAS = 000002
MS$SER  000002      MX$DAC  000020      N$LLT   000026      PR$BRD= 000040      RF.CTL= 000003
MS$SHU  000002      MX$DTE  000020      N$LLTM  000024      PR$BRU= 000020      RF.LD1= 000040
MS$SRV  000007      MX$FIL  000034      N$LVC   000036      PR$DWN= 000002      RF.LD2= 000100
MS$STA  000000      MX$LGC  000777      N$LV1   000002      PR$LCC= 000010      RF.TIM= 177400
MS$SYN  000012      MX$LIN  000020      N$LV2   000010      PR$MOP= 000004      RF.TMO= 000400
MS$TER  000001      MX$LON  000006      N$MBXQ  000050      PR$UP = 000001      RF.WFC= 000200
MS$TOP  000003      MX$NMS  000454      N$MHC1  000036      PS$CHR= 000016      RF.WTD= 000020
MS$TRI  000005      MX$NOD  000006      N$MHC2  000044      PS$FAI= 000014      RF.WTM= 000030
MS$UNR  000005      MX$OBJ  000006      N$PLD   000016      PS$NTI= 000006      RF.WTS= 000010
MS$VMS  000004      MX$OWN  000040      N$PLLT  000030      PS$OFF= 000000      RX.NXM= 000020
MS.DON  177600      MX$PAR  000036      N$PRI   000076      PS$STR= 000002      RX.OVR= 000010
MS.MOR  000002      MX$PAS  000010      N$RDA1  000022      PS$UP = 000012      R$R1  = ****** GX
MS.PAR  000003      MX$RAC  000047      N$RDA2  000030      PS$VER= 000010      R$R2  = ****** GX
MS.SUC  000001      MX$RID  000047      N$RTMX  000014      PS$WT = 000004      R$R3  = ****** GX
MTPCHA  001164R     MX$RPS  000047      N$RTM1  000014      PT$BRD= 000200      R$$DER= 000000
MT$AR4  000003      MX$SID  000040      N$RTM2  000015      PT$DRT= 000100      R$$K11= 000001
MT$ASC  000100      MX$SNK  000377      N$RT1   000000      PT$END= 000004      R$$SND= 000000
MT$BIL  000001      MX$TYP  000077      N$RT2   000006      PT$LV1= 000002      R$$11M= 000000
MT$BYS  000011      MX$UID  000020      N$SLA   000016      PT$LV2= 000001      R.CSR   000056
MT$CI   000007      M$$CRB= 000124      N$SNOD  000012      PT$PH3= 000010      R.FLG   000064
MT$CLE  000077      M$$CRX= 000000      N$TCTL  000112      PT$XAR= 000020      R.LST   000104
MT$COD  000200      M$$FCS= 000000      N$TIM   000004      PVCBLK= ****** GX   R.MAP   000110
MT$CON  000001      M$$MGE= 000000      N$TLC   000100      PVCCHA= ****** GX   R.QUE   000062
MT$DMC  000004      M$$NET= 000000      N$TNC   000114      PVCPRT  000010R     R.SLN   000070
MT$ETH  000006      M$$OVR= 000000      N$TRC   000106      PX$BLK= 000040      R.SRV   000060
MT$HEX  000040      NF$BLK= 000100      N$TTCB  000110      PX$DLM= 000200      R.STA   000066
```

```
**FILE**ID**CIRECO





  CCCCCCCC     IIIIII     RRRRRRRR     EEEEEEEEEE     CCCCCCCC     OOOOOO
  CCCCCCCC     IIIIII     RRRRRRRR     EEEEEEEEEE     CCCCCCCC     OOOOOO
CC               II       RR      RR   EE           CC           OO      OO
CC               II       RR      RR   EE           CC           OO      OO
CC               II       RR      RR   EE           CC           OO      OO
CC               II       RR      RR   EE           CC           OO      OO
CC               II       RRRRRRRR     EEEEEEEE     CC           OO      OO
CC               II       RRRRRRRR     EEEEEEEE     CC           OO      OO
CC               II       RR  RR       EE           CC           OO      OO
CC               II       RR  RR       EE           CC           OO      OO
CC               II       RR    RR     EE           CC           OO      OO         ....
CC               II       RR    RR     EE           CC           OO      OO         ....
  CCCCCCCC     IIIIII     RR      RR   EEEEEEEEEE     CCCCCCCC     OOOOOO           ....
  CCCCCCCC     IIIIII     RR      RR   EEEEEEEEEE     CCCCCCCC     OOOOOO           ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

G 14

CIRECO - READ/AND OR ZERO CIRCU MACRO V05.03b Saturday 29-Jun-85 12:17

```
Symbol table

MP$MRO   001605          MP$RET   001631          MP$USR   001750          MT$CON   000001          M$$MGE=  000000
MP$MRP   006322          MP$RFA   001323          MP$VEC   004410          MT$DMC   000004          M$$NET=  000000
MP$MRT   002153          MP$RMX   002201          MP$VER   004406          MT$ETH   000006          M$$OVR=  000000
MP$MRV   006320          MP$RPA   004374          MP$WDF   002165          MT$HEX   000040          NM.CLN   003400
MP$MSB   000170          MP$RPR   001606          MP$WMX   002177          MT$LPB   000005          NM.INI   002400
MP$MVE   000145          MP$RRT   002213          MP$XMT   002166          MT$MAX   000037          NM.OPR   003000
MP$MVI   001634          MP$RST   002212          MP$XPF   000242          MT$MUL   000100          NM.VR2   002401
MP$MVR   001751          MP$RSV   001754          MP$XPL   000244          MT$NLE   000017          NM.VR3   002402
MP$MWN   002154          MP$RSZ   001756          MP$XPT   000240          MT$NON   000001          N$$ACC=  000001
MP$MXB   002172          MP$RTI   001616          MP$XXX   177777          MT$NR4   000005          N$$BUF=  000001
MP$MXC   000466          MP$RTM   001611          MP$5ST   004406          MT$NTY   000060          N$$LDV=  000001
MP$MXR   001630          MP$RVE   001604          MS$ACT   000001          MT$OCT   000060          N$$MCP=  000001
MP$MXW   002143          MP$RVT   000156          MS$ADU   000010          MT$PHA   000002          N$$MLL=  000001
MP$NAA   006335          MP$SAC   000514          MS$ALD   000007          MT$POI   000000          N$$MOV=  000010
MP$NAC   001130          MP$SAD   000543          MS$ASE   000006          MT$QP2   000010          N$$NCT=  000001
MP$NAP   006326          MP$SCA   000310          MS$ATR   000011          MT$ROU   000000          N$$PEM=  000001
MP$NCT   000240          MP$SCO   000144          MS$AUT   000000          MT$R04   000004          PCLCIR=  ****** GX
MP$NET   002114          MP$SCT   002176          MS$CLE   000003          MT$SEC   000000          PDVTA =  ****** GX
MP$NLI   000765          MP$SDU   000203          MS$DED   000004          MT$SGD   000020          PVCCOU=  ****** GX
MP$NNA   000764          MP$SDV   000160          MS$DIE   000003          MT$SYS   000002          P$$P45=  000000
MP$NOD   000500          MP$SEH   004401          MS$DUM   000004          MT$TER   000001          P$$WRD=  000000
MP$NRA   006334          MP$SER   000144          MS$FAI   000013          MT$TRI   000002          Q$$OPT=  000010
MP$NRB   000214          MP$SET   000000          MS$GLO   000000          MT$TYP   007777          R$$DER=  000000
MP$NSA   001617          MP$SGZ   001644          MS$HOL   000002          MT$USD   000000          R$$K11=  000001
MP$NTI   002141          MP$SID   000176          MS$INA   000002          MT$X25   000003          R$$SND=  000000
MP$NUM   001642          MP$SIN   000310          MS$LOA   000003          MU$INC   000001          R$$11M=  000000
MP$NVE   001274          MP$SLI   000156          MS$LOO   000002          MU$OUT   000002          SCRATC   000000R
MP$NXN=  001476          MP$SLO   000171          MS$OFF   000001          MU$PER   000000          SF$ANM   000001
MP$OAC   000632          MP$SMX   002202          MS$ON    000000          MV$III   000000          SF$DNM   000002
MP$OAN   000620          MP$SND   000500          MS$PRO   000006          MV$IV    000001          SF.ACT=  000200
MP$OCO   000776          MP$SNP   006332          MS$REA   000004          MX$ACT   000020          SF.ENA=  000100
MP$OHO   000214          MP$SNU   000542          MS$REF   000001          MX$CIR   000020          SF.LPB=  000004
MP$ONA   000764          MP$SOB   000524          MS$RES   000003          MX$CLN   000020          SF.MFL=  000040
MP$ONR   002114          MP$SPA   000157          MS$RST   000001          MX$CNM   000006          SF.PAC=  000020
MP$OQL   001453          MP$SPR   000536          MS$RSX   000002          MX$CON   000006          SF.REA=  000010
MP$OTI   000777          MP$SPS   000513          MS$RT    000005          MX$DAC   000020          SF.SER=  000001
MP$OTY   001022          MP$STA   000000          MS$SER   000002          MX$DTE   000020          SF.SVC=  000002
MP$OUS   000777          MP$STI   002140          MS$SHU   000002          MX$FIL   000034          SF.UNL=  000040
MP$OVE   001010          MP$STT   002201          MS$SRV   000007          MX$LGC   000777          S$$WRG=  000000
MP$OWN   004374          MP$STY   000175          MS$STA   000000          MX$LIN   000020          S$$YSZ=  007600
MP$PAR   000036          MP$SUB   000001          MS$SYN   000012          MX$LON   000006          S.COST   000001
MP$PAS   000513          MP$SUR   000156          MS$TER   000001          MX$NMS   000454          S.FLG    000000
MP$PCH   002152          MP$SUS   000512          MS$TOP   000003          MX$NOD   000006          S.LEN    000004
MP$PCO   000024          MP$SVR   000163          MS$TRI   000005          MX$OBJ   000006          S.NMST   000002
MP$PCT   000144          MP$SWI   002260          MS$UNR   000005          MX$OWN   000040          S.OWNR   000003
MP$PDT   002114          MP$SYP   002424          MS$VMS   000004          MX$PAR   000036          T$$KMG=  000000
MP$PHA   000170          MP$TFL   000156          MS.DON   177600          MX$PAS   000010          T$$MIN=  000000
MP$PHY   000012          MP$TH1   002206          MS.MOR   000002          MX$RAC   000047          V$$CTR=  001000
MP$PLI   000025          MP$TH2   002205          MS.PAR   000003          MX$RID   000047          X$$DBT=  000000
MP$PLN   002140          MP$TH3   002202          MS.SUC   000001          MX$RPS   000047          ZF.COU=  001000
MP$PLO   000012          MP$TLN   000202          MT$AR4   000003          MX$SID   000040          ZF.DDM=  000001
MP$PMC   004374          MP$TLO   000172          MT$ASC   000100          MX$SNK   000377          ZF.DIA=  004000
MP$PNT   002126          MP$TPA   004375          MT$BIL   000001          MX$TYP   000077          ZF.DLC=  000002
MP$PRI   004411          MP$TRI   002164          MT$BYS   000011          MX$UID   000020          ZF.DVP=  100000
MP$PRO   002130          MP$TST   000144          MT$CI    000007          M$$CRB=  000124          ZF.INI=  040000
MP$PSS   001763          MP$TYP   002130          MT$CLE   000077          M$$CRX=  000000          ZF.KMX=  000020
MP$PST   001762          MP$UCS   004407          MT$COD   000200          M$$FCS=  000000          ZF.LLC=  000004
```

```
                                                                  F 16
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17  Page 6-2
Symbol table

A$$CHK= 000000              G$$TSS= 000000              LP$WDV= 000020              MC$CAP  001440              MC$SYS  000024
A$$CPS= 000000              G$$TTK= 000000              LP$WLD= 000360              MC$CCL  001445              MC$SYZ  000000
A$$PRI= 000000              G$$WRD= 000000              LP$WTR= 000200              MC$CDC  002045              MC$TBR  001140
A$$TRP= 000000              HEADR = ****** GX           LP$WUN= 000100              MC$CDP  001441              MC$TBS  001141
BIT0  = 000001              H.CSP   000000              L$ADJ   000110              MC$CIF  001465              MC$TDR  002053
BIT1  = 000002              H.CUIC  000010              L$BUF   000060              MC$CLD  001464              MC$TIM  000000
BIT10 = 002000              H.DSW   000046              L$CHN   000054              MC$COU  100000              MC$TYP  007777
BIT11 = 004000              H.DUIC  000012              L$COU   000114              MC$CTL  001454              MC$UBU  002052
BIT12 = 010000              H.EFLM  000004              L$CTB   000053              MC$CTR  001452              MC$UFD  002047
BIT13 = 020000              H.EFSV  000040              L$CTL   000002              MC$CTS  001453              MC$UMR  001132
BIT14 = 040000              H.FCS   000050              L$DDM   000000              MC$DOV  002050              MC$UMS  001133
BIT15 = 100000              H.FORT  000052              L$FLG   000064              MC$LBR  001750              MC$VAX  000003
BIT2  = 000004              H.FPSA  000042              L$FLX   000070              MC$LBS  001751              MC$WID  060000
BIT3  = 000010              H.FPSL= 000062              L$LEN   000124              MC$LDI  001774              MC$WIH  040000
BIT4  = 000020              H.FPVA  000034              L$LTM   000070              MC$LDO  001775              MC$WIL  020000
BIT5  = 000040              H.GARD  000072              L$MSG   000056              MC$LDR  001762              MC$W08  020000
BIT6  = 000100              H.HDLN  000002              L$MTYP  000067              MC$LDS  001763              MC$W16  040000
BIT7  = 000200              H.IPC   000016              L$NAM   000000              MC$LLB  002021              MC$W32  060000
BIT8  = 000400              H.IPS   000014              L$NLEN= 000020              MC$LLP  002115              MC$XBR  001750
BIT9  = 001000              H.ISP   000020              L$NOD   000112              MC$LLR  002007              MC$XBS  001751
B$10    000003              H.LUN   000076              L$NXT   000044              MC$LOD  000001              MC$XCJ  000322
B$11    000004              H.NLUN  000074              L$OPT   000062              MC$LRB  002020              MC$XCR  002260
B$12    000005              H.NML   000061              L$PAR   000120              MC$LRP  002114              MC$XCS  002261
B$5     000000              H.ODVA  000022              L$PDV   000055              MC$LRR  002006              MC$XDR  001762
B$6     000001              H.ODVL  000024              L$PFG   000041              MC$LSI  002032              MC$XDS  001763
B$7     000002              H.OVLY  000054              L$PLB   000116              MC$LST  002033              MC$XFR  002272
B$7S10  000006              H.PFVA  000032              L$PRO   000067              MC$LUP  006072              MC$XFS  002273
CHKCIR= ****** GX           H.RCVA  000036              L$PVC   000004              MC$MAP  010000              MC$XLJ  000323
CIREIN  000000RG            H.RRVA  000062              L$SCN   000020              MC$MBL  001764              MC$XLR  002330
C$$CKP= 000000              H.SPRI  000060              L$SCR   000072              MC$MBX  004420              MC$XMA  000310
C$$ORE= 000400              H.TKVA  000026              L$SLEN  000066              MC$MBY  001752              MC$XMC  002305
C$$RSH= 177564              H.TKVL  000030              L$SLT   000042              MC$MPX  004406              MC$XMS  002304
DV.CCL= 000002              H.VEXT  000056              L$SNM   000046              MC$NAP  001604              MC$XNR  002332
DV.COM= 020000              H.WND   000044              L$SYL   000122              MC$NBR  001130              MC$XRC  002316
DV.DIR= 000010              H.X25   000064              L$TNM   000052              MC$NBS  001131              MC$XRR  002331
DV.EXT= 000400              I$$RAR= 000000              L$TPT   000050              MC$NCR  001154              MC$XRS  002342
DV.F11= 040000              I$$RDN= 000000              L$TRB   000040              MC$NCS  001155              MC$020  000002
DV.ISP= 002000              K$$CNT= 177546              L$TSZ   000070              MC$NML  001274              MD$CI   000007
DV.MBC= 000400              K$$CSR= 177546              L$TYP   000063              MC$NMR  001142              MD$CNA  000003
DV.MNT= 100000              K$$LDC= 000000              L$UNT   000003              MC$NMS  001143              MD$COU  000001
DV.MSD= 000100              K$$TPS= 000074              L$$ASG= 000000              MC$NND  001606              MD$DA   000010
DV.OSP= 004000              LC$EXT  000000              L$$DRV= 000000              MC$NNU  001605              MD$DL   000004
DV.PSE= 010000              LC$INT  000001              L$$P11= 000001              MC$NOP  001607              MD$DLV  000020
DV.REC= 000001              LD$LP = 000000              L$$11R= 000000              MC$NOR  000000              MD$DMC  000014
DV.SDI= 000020              LF$END= 000020              MB$DIS  000001              MC$NPF  001616              MD$DMF  000046
DV.SQD= 000040              LF$MLT= 040000              MB$ENA  000000              MC$NPR  001630              MD$DMP  000022
DV.SWL= 001000              LF$REA= 000001              MB$FUL  000002              MC$NRE  001200              MD$DMR  000050
DV.TTY= 000004              LF$SEG= 100000              MB$MIX  000002              MC$NRT  001166              MD$DMV  000042
DV.UMD= 000200              LF$SIG= 000040              MB$ONE  000001              MC$NVR  001642              MD$DN   000016
D$$BUG= 177514              LF$SKP= 000004              MB$RXO  000001              MC$P08  000000              MD$DP   000000
D$$ISK= 000000              LF$VR2= 000010              MB$TXO  000000              MC$P11  000001              MD$DPV  000044
D$$L11= 000001              LF$ZER= 000002              MB$ZER  000000              MC$RFL  002046              MD$DQ   000006
D$$YNC= 000000              LP$MPT= 000010              MC$BAB  006073              MC$SBU  002051              MD$DTE  000024
D$$YNM= 000000              LP$MUX= 000004              MC$BID  001765              MC$SFL  002044              MD$DU   000002
E$$XPR= 000000              LP$TRB= 000002              MC$BMC  001767              MC$SYC  000012              MD$DUP  000012
F$$LVL= 000001              LP$UNT= 000001              MC$BSC  001766              MC$SYL  000036              MD$DV   000026
G$$TPP= 000000              LP$WCN= 000040              MC$CAC  001442              MC$SYR  000050              MD$DZ   000030
```

```
                                                                  G 16
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17  Page 6-3
```

```
                                                                     G  1
CICHIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 1

SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL   VALUE               REFERENCES

ACTACT     000544 R          #17-810
BBTACT     001210 R          #29-1088
BSAACT     001246 R          #32-1142
BSDACT     001260 R          #33-1160
BSIACT     001272 R          #34-1178
CALL       002024 R            -483
CALL1      002030 R          #9-486
CHKCIR  =  ******  GX         10-546
CHKDMP     000606 R          #19-856     28-1070     29-1088     30-1106     31-1124     32-1142     33-1160     34-1178     35-1196
                              36-1214    37-1232     38-1250     39-1268     40-1286     41-1304
CHNACT     000662 R          #20-884
CICHIN     000000 RG         #10-528
CIRERR     000476 R          #15-757
 LCT       002036 R          #9-491
CLECIR     000422 R          #8-193      10-560
CMBACT     000710 R          #21-904
COSACT     000170 R          #11-607
CRET       002044 R          #9-496
CSTERR     000512 R          #15-762
DTEACT     000726 R          #22-924
END        000450 R          #8-206
FLG$$$  =  177777             #8-82      #8-86       8-88      #8-88       8-90      #8-90      #8-92      #8-93       8-94
                              #8-94      #8-95      #8-96       8-97      #8-97      #8-98      #8-99      8-100     #8-100
                             #8-101     #8-102      8-103     #8-103     #8-104     #8-105      8-106     #8-106     #8-107
                             #8-108      8-109     #8-109     #8-110     #8-111      8-112     #8-112     #8-113     #8-114
                              8-115     #8-115     #8-116     #8-117      8-118     #8-118     #8-119     #8-120      8-121
                             #8-121     #8-122     #8-123      8-124     #8-124     #8-125     #8-126      8-127     #8-127
                             #8-128     #8-129      8-130     #8-130     #8-131     #8-132      8-133     #8-133     #8-134
                             #8-135      8-136     #8-136     #8-137     #8-138      8-139     #8-139     #8-140     #8-141
                              8-142     #8-142     #8-143     #8-144      8-145     #8-145     #8-146     #8-147      8-148
                             #8-148     #8-149     #8-150      8-151     #8-151     #8-152     #8-153      8-154     #8-154
                             #8-155     #8-156      8-157     #8-157     #8-158     #8-159      8-160     #8-160     #8-161
                             #8-162      8-163     #8-163     #8-164     #8-165      8-166     #8-166     #8-167     #8-168
                              8-169     #8-169     #8-170     #8-171      8-172     #8-172     #8-173     #8-174      8-175
                             #8-175     #8-176     #8-177      8-178     #8-178     #8-179     #8-180      8-181     #8-181
                             #8-182     #8-183      8-184     #8-184     #8-185     #8-186      8-187     #8-187     #8-188
                             #8-193      8-194     #8-194     #8-195     #8-196      8-197     #8-197     #8-198     #8-199
                              8-200     #8-200     #8-201     #8-206      8-208     #8-208     #9-215     #9-218      9-219
                             #9-219     #9-225      9-226     #9-226     #9-227      9-228     #9-228      9-229     #9-229
                              9-231     #9-231     #9-238      9-239     #9-239     #9-240     #9-245      9-246     #9-246
                             #9-247      9-248     #9-248      9-249     #9-249     #9-253      9-254     #9-254     #9-255
                              9-256     #9-256     #9-261      9-262     #9-262     #9-263      9-264     #9-264     #9-269
                              9-270     #9-270     #9-271     #9-276      9-277     #9-277     #9-278      9-279     #9-279
                              9-280     #9-280     #9-281      9-282     #9-282     #9-284      9-285     #9-285     #9-287
                              9-288     #9-288     #9-292      9-293     #9-293     #9-294     #9-299      9-300     #9-300
                             #9-301     #9-306      9-307     #9-307     #9-308      9-309     #9-309      9-310     #9-310
                             #9-315      9-316     #9-316     #9-317      9-318     #9-318     #9-322      9-323     #9-323
                             #9-324      9-325     #9-325     #9-329      9-330     #9-330     #9-331      9-332     #9-332
                             #9-336      9-337     #9-337     #9-338      9-339     #9-339     #9-343      9-344     #9-344
                             #9-345      9-346     #9-346     #9-350      9-351     #9-351     #9-352      9-353     #9-353
                             #9-357      9-358     #9-358     #9-359      9-360     #9-360     #9-364      9-365     #9-365
                             #9-366      9-367     #9-367     #9-371      9-372     #9-372     #9-373      9-374     #9-374
```

```
                                                                     H  1
CICHIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 2
```

Table of contents

```
     6-    59    MACRO CALLS AND LOCAL DATA
     7-    71    Context area definition
     8-    75    DISPATCH TABLE FOR OPERATE ROUTINES
     9-   130    CICHOP - CIRCUIT CHANGE PARAMETERS OPERATE
    10-   263    DSPTCH - DISPATCH TO PROPER ROUTINE
    11-   292    SETUP - SET UP FOR COMMAND PROCESSING
    12-   335    CLNUP - CLEAN UP AFTER AN OPERATION FAILED
    13-   372    FNDCTR -  FIND COUNTER BLOCK
```

```
                                                .SBTTL  CLNUP - CLEAN UP AFTER AN OPERATION FAILED
                                        ;+
                                        ; **-CLNUP-CLEAN UP AFTER AN OPERATION HAS FAILED
                                        ;
                                        ; THIS ROUTINE IS CALLED TO RESTORE A CIRCUIT TO IT'S INITIAL STATE
                                        ; AFTER AN OPERATION HAS FAILED.
                                        ;
                                        ; INPUTS:
                                        ;       R0 = ERROR CODE FROM FAILURE
                                        ;       R4 = ADDRESS OF CONTEXT BLOCK
                                        ;       R5 = POINTER INTO DISPATCH TABLE WITH:
                                        ;            -4(R5) = BIT MASK FOR OPERATION THAT FAILED
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CIRCUIT HAD BEEN LOADED, IT IS CLEARED
                                        ;
                                        ; REGISTERS:
                                        ;       R0, R4, R5 ARE PRESERVED
                                        ;-

000552                                  CLNUP:
000552                                          SAVRG   <R0>                    ; SAVE REAL ERROR CODE
000554  022765  000001  177774                  CMP     #LS$NTL,-4(R5)          ; WAS THE OPERATION THAT FAILED AN NTL SET ?
000562  001415                                  BEQ     10$                     ; IF EQ, YES - NOTHING TO CLEANUP
000564  032764  000001  000070                  BIT     #LS$NTL,L$FL2(R4)       ; WAS CIRCUIT LOADED IN THIS COMMAND ?
000572  001411                                  BEQ     10$                     ; IF EQ, NO - DON'T CLEAR IT
000574  016446  000134                          MOV     L$NTL(R4),-(SP)         ; ELSE, SAVE NTL FUNCTION WORD
000600  012764  000006  000134                  MOV     #LX.LIN,L$NTL(R4)       ; SET "CLEAR LINE" FUNCTION
000606                                          CALL    CLENTL                  ; TRY TO CLEAR THE LINE
000612  012664  000134                          MOV     (SP)+,L$NTL(R4)         ; RESTORE NTL FUNCTION WORD
000616                                  10$:    RESRG   <R0>                    ; RESTORE PREVIOUS ERROR CODE
000620                                          RETURN

                                                .ENDC ; NDF R$$11S & S$$BAS & R$$RTR

                                                .IF NDF R$$11S
```

```
                                                          G  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13      PAGE 6
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE              REFERENCES

M$$MGE  = 000000            13-418
NXTCIR  = ******   GX       9-184
N$TLC     000100            13-411    13-415    13-416
N$$VCT  = ******            13-409    13-415
PDVTA   = ******   GX       13-397
R$$PRO  = ******            7-73      7-73      8-97      8-111     9-163     9-175     9-207     9-225     9-240
R$$RTR  = ******            8-85      8-97      8-111     10-287    11-333
R$$11D  = ******            6-67
R$$11M  = 000000            6-67      13-418
R$$11S  = ******            6-67      8-77      8-84      8-107     9-158     9-175     9-192     9-222     10-287
                            11-312    11-333    12-370
SETACT  = ******   GX       8-92
SETBBT  = ******   GX       8-99
SETBSA  = ******   GX       8-112
SETBSD  = ******   GX       8-113
SETBSI  = ******   GX       8-114
SETCOS  = ******   GX       8-88
SETHTM  = ******   GX       8-124
SETINA  = ******   GX       8-115
SETIND  = ******   GX       8-116
SETINI  = ******   GX       8-117
SETMXB  = ******   GX       8-121
SETNMT  = ******   GX       8-100
SETNTL  = ******   GX       8-86
SETOWN  = ******   GX       8-89
SETSER  = ******   GX       8-103
SETSTA  = ******   GX       8-95
SETTAD  = ******   GX       8-91
SETTH1  = ******   GX       8-118
SETTH2  = ******   GX       8-119
SETTH3  = ******   GX       8-120
SETUP     000476 R          9-186    #11-311
SETXMT  = ******   GX       8-101
SF.ACT  = 000200           #6-68
SF.ENA  = 000100           #6-68
SF.LPB  = 000004           #6-68
SF.MFL  = 000040           #6-68
SF.PAC  = 000020           #6-68
SF.REA  = 000010           #6-68
SF.SER  = 000001           #6-68
SF.SVC  = 000002           #6-68
SF.UNL  = 000040           #6-68
SLTMA   = ******   GX       13-402
S$$BAS  = ******            8-85      8-97      8-111     10-287    11-333
S.COST    000001           #6-68
S.FLG     000000           #6-68
S.LEN     000004           #6-68
S.NMST    000002           #6-68
S.OWNR    000003           #6-68      13-396
T$FLAG    000044           #6-64
T$LIF     000013           #6-64
T$LIFL    000013           #6-64
```

```
                                                          H  4
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13      PAGE 7
```

```
                                                .SBTTL  SET DMP PARAMETERS TABLES

                                        ;
                                        ; THESE TABLES ARE USED BY THE DMP SET ROUTINE TO SET
                                        ; THE SPECIFIED PARAMETER IN THE DMP LINE TABLE.  EACH
                                        ; TABLE HAS THE FOLLOWING FORMAT:
                                        ;
                                        ; PARMVALUE, TRIB FLAG, CONTEXT AREA OFFSET, LINE TABLE OFFSET, TYPE FLAG
                                        ;
                                        ;
                                        ; WHERE EACH ENTRY IS A WORD.
                                        ; TRIB FLAG = 0 IF PARAMETER IS APPLIED ACROSS ALL TRIBUTARIES
                                        ;           = 1 IF PARAMETER IS APPLIED BY TRIBUTARY
                                        ; TYPE FLAG = 0 IF PARAMETER IS A WORD VALUE
                                        ;             1 IF PARAMETER IS A BYTE VALUE
                                        ; BIT       = BIT TO SET IN POLLING FLAG WORD TO INDICATE THAT THE PARAMETER
                                        ;             HAS BEEN CHANGED
                                        ;
 000120  002165  000001  000114 BBTTBL: .WORD   MP$BBT, 1, L$BBT, T.BTTC+0, 0
 000126  000074  000000
 000132  002141  000001  000116 NMTTBL: .WORD   MP$NTI, 1, L$NMT, T.SITC+0, 0
 000140  000072  000000
 000144  002166  000001  000120 XMTTBL: .WORD   MP$XMT, 1, L$XMT, T.DELT+0, 0
 000152  000056  000000
 000156  002176  000001  000122 BSATBL: .WORD   MP$BSA, 1, L$BSA, T.QRAC+0, 1
 000164  000060  000001
 000170  002203  000001  000123 BSDTBL: .WORD   MP$BSD, 1, L$BSD, T.QRPD+0, 1
 000176  000064  000001
 000202  002200  000001  000124 BSITBL: .WORD   MP$BSI, 1, L$BSI, T.QRIN+0, 1
 000210  000062  000001
 000214  002177  000001  000125 INATBL: .WORD   MP$INA, 1, L$INA, T.QRAC+1, 1
 000222  000061  000001
 000226  002204  000001  000126 INDTBL: .WORD   MP$IND, 1, L$IND, T.QRPD+1, 1
 000234  000065  000001
 000240  002201  000001  000127 INITBL: .WORD   MP$INI, 1, L$INI, T.QRIN+1, 1
 000246  000063  000001
 000252  002206  000001  000130 TH1TBL: .WORD   MP$TH1, 1, L$TH1, T.TOD +0, 1
 000260  000070  000001
 000264  002205  000001  000131 TH2TBL: .WORD   MP$TH2, 1, L$TH2, T.NDM +1, 1
 000272  000067  000001
 000276  002202  000001  000132 TH3TBL: .WORD   MP$TH3, 1, L$TH3, T.NDM +0, 1
 000304  000066  000001
 000310  002172  000001  000133 MXBTBL: .WORD   MP$MXB, 1, L$MXB, T.TOD +1, 1
 000316  000071  000001

                                                .ENDC   ; NDF R$$PRO

```

```
002052  012700  177747          30$:    MOV     #ME.OPE,R0              ; ASSUME OPERATION FAILED
002056  103406                          BCS     90$                     ; IF CS, IT FAILED
                                ;
                                ; SET NEW STATE TO SERVICE
                                ;
002060  142712  000017          40$:    BICB    #17,(R2)                ; CLEAR PREVIOUS STATE BITS
002064  152712  000002                  BISB    #MS$SER,(R2)            ; SET NEW STATE AS SERVICE
002070  000241                          CLC                             ;
002072  000401                          BR      100$                    ; AND RETURN

002074  000261                  90$:    SEC
002076                          100$:   RETURN
```

```
                                        .SBTTL  SETSER - Set SERVICE [ENABLE/DISABLE]
                                ;+
                                ; ** SETSER - Set SERVICE [ENABLE/DISABLE]
                                ;
                                ; Inputs:       R4 = context area address
                                ;               L$SER(R4) = coded value for service
                                ;
                                ; Output:       Service is enabled/disabled for the circuit.
                                ;-

000566                          SETSER::
                                        .IIF NE MB$ENA  .ERROR          ; Value error

000566  026427  000000  015355          CMP     L$DDM(R4),#^RDLM        ; DLM circuit ?
000574  001437                          BEQ     15$                     ; if EQ, yes- parameter not applicable
000576  022764  177777  000050          CMP     #-1,L$TPT(R4)           ; PVC ?
000604  001433                          BEQ     15$                     ; yes

000606  016400  000042                  MOV     L$SLT(R4),R0            ; get SLT address
000612  105760  000014                  TSTB    L.NSTA(R0)              ; Multipoint/Ethernet ?
000616  001012                          BNE     10$                     ; if NE,yes
000620  042760  000040  000000          BIC     #LF.SER,L.FLG(R0)       ; assume service enabled
000626  005764  000230                  TST     L$SER(R4)               ; disable service ?
000632  001427                          BEQ     20$                     ; if NE, no
000634  052760  000040  000000          BIS     #LF.SER,L.FLG(R0)       ; disable service on circuit
000642  000423                          BR      20$                     ; done

000644  016400  000050          10$:    MOV     L$TPT(R4),R0            ; get trib pointer
000650  142760  000001  000000          BICB    #SF.SER,S.FLG(R0)       ; assume service enabled
000656  005764  000230                  TST     L$SER(R4)               ; disable service ?
000662  001413                          BEQ     20$                     ; if EQ, no
000664  152760  000001  000000          BISB    #SF.SER,S.FLG(R0)       ; disable service
000672  000407                          BR      20$                     ; done

000674  012700  177752          15$:    MOV     #ME.PNA,R0              ; else parameter not applicable,
000700  012764  000144  000072          MOV     #MP$SER,L$PAR(R4)       ; service
000706  000261                          SEC                             ; error
000710  000400                          BR      20$                     ; done

000712                          20$:    RETURN

                                        .ENDC   ; NDF R$$11S

        000001                          .END
```

```
                                                                         G  8
CICHPD      CREATED BY  MACRO   ON 29-JUN-85 AT 12:14       PAGE 6
SYMBOL CROSS REFERENCE                                      CREF    04.00
SYMBOL   VALUE             REFERENCES
L$TH3      000132          #6-86       8-141
L$TNM      000052          #6-86       10-203
L$TPT      000050          #6-86       12-286      13-336      13-371      18-731      19-812      20-859      20-881      21-956
                            21-974     23-1078     24-1124     24-1136
L$TRB      000040          #6-86       9-167       18-761      23-1083
L$TYP      000063          #6-86
L$UNT      000003          #6-86       11-230
L$XCH      000232          #6-86
L$XMT      000120          #6-86       8-132
L.COST     000015          #5-73      *20-876
L.CTL      000012          #5-73
L.CVA      177776          #5-73
L.DDM      000002          #5-73
L.DDS      000004          #5-73
L.DLC      000003          #5-73
L.DLM      000006          #5-73
L.DLS      000010          #5-73
L.FLG      000000          #5-73       9-172       12-291      13-348      13-358      13-366      13-439      13-441      15-596
                          *18-766      19-810     *19-825      20-869      20-877      21-976      21-980     *24-1130    *24-1133
L.KRBA     000016          #5-73
L.LEN    = 000022          #5-73
L.MPF      000022          #5-73       13-443     *18-765      19-816
L.NMST     000020          #5-73       13-363      15-596      18-734     *21-1000
L.NSTA     000014          #5-73       13-356      18-729      18-759      19-814      20-873      21-978      21-998      24-1128
L.OWNR     000021          #5-73
L.UNT      000013          #5-73
MB$ENA     000000           24-1120
ME$CIR     000003           9-175      13-347      13-414      14-546      15-619      17-694      18-727      20-862      21-959
ME$LIN     000001           12-290
ME.CST     177765           9-174      12-289      13-346      13-413      14-545      15-618      17-693      18-726      20-868
                            21-975
ME.FUN     177777           21-967
ME.IID     177767           13-365     20-861      21-958
ME.OPE     177747           11-258     13-461      14-565      15-633      16-668      17-702      21-1004
ME.PNA     177752           12-284     12-293      13-338      20-857      21-955      23-1076     24-1143
ME.PVA     177760           15-600
MP$BBT     002165           8-130
MP$BSA     002176           8-133
MP$BSD     002203           8-134
MP$BSI     002200           8-135
MP$COS     001604           20-858
MP$HTM     001612           12-285     12-294
MP$INA     002177           8-136
MP$IND     002204           8-137
MP$INI     002201           8-138
MP$MXB     002172           8-142
MP$NTI     002141           8-131
MP$SER     000144           24-1144
MP$STA     000000           13-339     15-601
MP$TH1     002206           8-139
MP$TH2     002205           8-140
MP$TH3     002202           8-141
```

```
                                                                         H  8
CICHPD      CREATED BY  MACRO   ON 29-JUN-85 AT 12:14       PAGE 7
```

```
    107                                         .SBTTL  Control block parameter format tables
    108                                 ;
    109                                 ; Each table corresponds to a particular control block format. The flags word
    110                                 ; is either 0 (denoting an unused or ignored field in the block) or
    111                                 ; is used to control the copying of fields from the SOURCE (the
    112                                 ; context area) to the TARGET (the control block to be changed).
    113                                 ;
    114                                 ; FORMAT OF FORMAT MAPPING TABLES IS AS FOLLOWS:
    115                                 ;
    116                                 ;       Flag    Source  Source  Source  Target          Target  Target
    117                                 ;       Bit     Field   Length  Length  Field           Length  Length
    118                                 ;                       Parm.   Routine Offset          Parm.   Routine
    119                                 ;
    120                                 ; Define SVC descriptor
    121
    122 000000                          SVCMAP:
    123 000000                           PARMAP LS$CMB, L$CMB,  2,      0,      S$PKSZ,         2,      0
    124 000016                           PARMAP LS$MXW, L$MXW,  1,      0,      S$WND-S$PKSZ,   1,      0
    125 000034                           PARMAP LS$NUM, L$NUML, 1,      0,      S$ADSZ-S$WND,   1,      0
    126 000052                           PARMAP LS$MXR, L$MXR,  1,      0,      S$LFT-S$ADSZ,   1,      0
    127 000070                           PARMAP LS$MXR, L$MXR,  1,      0,      S$RCL-S$LFT,    1,      0
    128 000106                           PARMAP LS$RET, L$RET,  2,      0,      S$TIM-S$RCL,    2,      0
    129 000124                           PARMAP LS$RET, L$RET,  2,      0,      S$TMR-S$TIM,    2,      0
    130 000142                           PARMAP LS$NUM, L$NUM,  8.,     0,      S$DTE-S$TMR,    8.,     0
    131 000160                           PARMAP 0,      0,      0,      0,      S$LEN,          0,      0
    132 000176                           PARMAP
```

```
Symbol table

MS.PAR  000003        MX$RID  000047        PT$BRO= 000200        R$$SND= 000000        T$LTIM  000026
MS.SUC  000001        MX$RPS  000047        PT$DRT= 000100        R$$11M= 000000        T$LTPR  000014
MT$AR4  000003        MX$SID  000040        PT$END= 000004        SETCTR  000662R       T$LTPS  000020
MT$ASC  000100        MX$SNK  000377        PT$LV1= 000002        SETLCT  001070R       T$NAPL  000004
MT$BIL  000001        MX$TYP  000077        PT$LV2= 000001        SETPVC  000326R       T$NFE   000000
MT$BYS  000011        MX$UID  000020        PT$PH3= 000010        SETSVC  000400R       T$NLEN  000010
MT$CI   000007        M$$CRB= 000124        PT$XAR= 000020        SF$ANM  000001        T$NNUL  000002
MT$CLE  000077        M$$CRX= 000000        PX$BLK= 000040        SF$BLK= 000001        T$NOPL  000006
MT$COD  000200        M$$FCS= 000000        PX$DLM= 000200        SF$DNM  000002        T$NRNI  000042
MT$CON  000001        M$$MGE= 000000        PX$SVC= 000100        SF$INC= 000002        T$NRPL  000005
MT$DMC  000004        M$$NET= 000000        P$CHN   000004        SF.ACT= 000200        T$NRUL  000007
MT$ETH  000006        M$$OVR= 000000        P$CNT   000005        SF.ENA= 000100        T$NVR   000001
MT$HEX  000040        NM.CLN  003400        P$CTR   000034        SF.LPB= 000004        T$RPRI  000040
MT$LPB  000005        NM.INI  002400        P$DRC1  000015        SF.MFL= 000040        T$SVC   000034
MT$MAX  000037        NM.OPR  003000        P$DRTR  000036        SF.PAC= 000020        T$T5    000030
MT$MUL  000100        NM.VR2  002401        P$FLG   000012        SF.REA= 000010        T$T6    000032
MT$NLE  000017        NM.VR3  002402        P$FWD   000030        SF.SER= 000001        T$$KMG= 000000
MT$NON  000001        N$$ACC= 000001        P$ICCB  000046        SF.SVC= 000002        T$$MIN= 000000
MT$NR4  000005        N$$BUF= 000001        P$IPL   000014        SF.UNL= 000040        VF$RCV= 100000
MT$NTY  000060        N$$LDV= 000001        P$LCD   000002        SVCFLG  000720R       VF$XMT= 040000
MT$OCT  000060        N$$MCP= 000001        P$LEN   000052        SVCMAP  000000R       V$FLG   000000
MT$PHA  000002        N$$MLL= 000001        P$LST   000000        S$ADSZ  000007        V$LEN   000022
MT$POI  000000        N$$MOV= 000010        P$NRNI  000040        S$CTL   000002        V$RCV   000002
MT$QP2  000010        N$$NCT= 000001        P$OCCB  000050        S$DTE   000020        V$XMT   000012
MT$ROU  000000        N$$PEM= 000001        P$PFQ   000006        S$FLG   000010        V$$CTR= 001000
MT$RO4  000004        PF$BLK= 020000        P$PKSZ  000044        S$LEN   000032        XA$NCO= 000020
MT$SEC  000000        PF$CLC= 010000        P$RMX1  000016        S$LFT   000012        XA$RR = 000200
MT$SGD  000020        PF$DLM= 100000        P$RMX2  000020        S$LNK   000000        XA$UCL= 000040
MT$SYS  000002        PF$EIP= 000002        P$RPRI  000042        S$OWNR  000030        XA$URE= 000100
MT$TER  000001        PF$ENB= 000001        P$RTIM  000003        S$PKSZ  000004        XF$NCO= 000020
MT$TRI  000002        PF$FAI= 004000        P$STA1  000022        S$RCL   000013        XF$OFF= 000200
MT$TYP  007777        PF$FM1= 000100        P$STA2  000023        S$STA   000003        XF$ORI= 000002
MT$USD  000000        PF$FM2= 000200        P$TIM   000010        S$TIM   000014        XF$OTI= 000001
MT$X25  000003        PF$OFF= 000000        P$TSCT  000026        S$TMR   000016        XF$TIL= 000010
MU$INC  000001        PF$RM1= 000020        P$TSIZ  000024        S$WND   000006        XF$TIR= 000004
MU$OUT  000002        PF$RM2= 000040        P$TYP   000001        S$$WRG= 000000        XT$CHN= 000100
MU$PER  000000        PF$RVR= 000010        P$$P45= 000000        S$$YSZ= 007600        XT$DGR= 000002
MV$III  000000        PF$STA= 000004        P$$WRD= 000000        S.COST  000001        XT$FAS= 000200
MV$IV   000001        PF$SVC= 040000        Q$$OPT= 000010        S.FLG   000000        XT$INC= 000004
MX$ACT  000020        PF$UP = 000004        RF.CTL= 000003        S.LEN   000004        XT$OUC= 000010
MX$CIR  000020        PR$BED= 000200        RF.LD1= 000040        S.NMST  000002        XT$PVC= 000001
MX$CLN  000020        PR$BEU= 000100        RF.LD2= 000100        S.OWNR  000003        X$ABQ   000076
MX$CNM  000006        PR$BRD= 000040        RF.TIM= 177400        T$FLAG  000044        X$ALF   000055
MX$CON  000006        PR$BRU= 000020        RF.TMO= 000400        T$LIF   000013        X$AUC   000056
MX$DAC  000020        PR$DWN= 000002        RF.WFC= 000200        T$LIFL  000013        X$CLEN= 000025
MX$DTE  000020        PR$LCC= 000010        RF.WTD= 000020        T$LIFO  000013        X$CTIM  000060
MX$FIL  000034        PR$MOP= 000004        RF.WTM= 000030        T$LIFS  000013        X$DIAG  000003
MX$LGC  000777        PR$UP = 000001        RF.WTS= 000010        T$LIN   000000        X$DTE   000024
MX$LIN  000020        PSIPT = ****** GX     R$PC  = 000016        T$LIPS  000006        X$FLG   000014
MX$LON  000006        PS$CHR= 000016        R$R0  = 000002        T$LLD   000012        X$GLEN  000106
MX$NMS  000454        PS$FAI= 000014        R$R1  = 000004        T$LLDC  000045        X$LCN   000026
MX$NOD  000006        PS$NTI= 000006        R$R2  = 000006        T$LLDL  000012        X$LEN   000106
MX$OBJ  000006        PS$OFF= 000000        R$R3  = 000010        T$LLDO  000012        X$MOWN  000023
MX$OWN  000040        PS$STR= 000002        R$R4  = 000012        T$LLDS  000012        X$NLRE  000052
MX$PAR  000036        PS$UP = 000012        R$R5  = 000014        T$LLEN  000046        X$NNRE  000054
MX$PAS  000010        PS$VER= 000010        R$$DER= 000000        T$LOPR  000002        X$NPL   000006
MX$RAC  000047        PS$WT = 000004        R$$K11= 000001        T$LTCL  000024        X$NPR   000007
```

```
    000063                          L$TYP:  .BLKB   1                       ; LINE-ID FORMAT TYPE
    000064                          L$FLG:  .BLKW   1                       ; FLAG WORD
    000066                          L$SLEN: .BLKB   1                       ; Significant length of circuit name
    000067                          L$PRO:                                  ; Line protocol
    000067                          L$MTYP: .BLKB   1                       ; Network management circuit type
    000070                          L$TSZ:                                  ; Block size for transport circuits
    000070                          L$LTM:                                  ; Listen timer for transport circuits
    000070                          L$FLX:  .BLKW   1                       ; Flags word for X.25 circuit commands
    000072                          L$SCR:  .BLKW   7                       ; SCRATCH BUFFER
    000110                          L$ADJ:  .BLKW   1                       ; Pointer to adjacency database
    000112                          L$NOD:  .BLKW   1                       ; Adjacent node address
    000114                          L$COU:  .BLKW   1                       ; Count of adjacency entries
    000116                          L$PLB:  .BLKW   1                       ; PLB address
    000120                          L$PAR:  .BLKW   1                       ; PARAMETER TYPE
    000122                          L$SYL:  .BLKW   1                       ; System line number
    000124                          L$XLEN:                                 ; LENGTH OF CONTEXT AREA
    000000                                  .PSECT
                                    ;
                                    ; OFFSETS INTO SCRATCH BUFFER (L$SCR) FOR DMC BASE TABLE COUNTERS
                                    ;
    000124                                  .ASECT
                000000                      .=0
    000000                          B$5:    .BLKB   1                       ; BASE TABLE + 5
    000001                          B$6:    .BLKB   1                       ; BASE TABLE + 6
    000002                          B$7:    .BLKB   1                       ; BASE TABLE + 7
    000003                          B$10:   .BLKB   1                       ; BASE TABLE + 10
    000004                          B$11:   .BLKB   1                       ; BASE TABLE + 11
    000005                          B$12:   .BLKB   1                       ; BASE TABLE + 12
    000006                          B$7S10: .BLKB   1                       ; SUM OF BASE TABLE 7 AND 10
                                            .EVEN
    000000                                  .PSECT
    543                                             ;
    544                                             ; Non-Ethernet circuits (Router): return listen timer for single adjacency
    545                                             ;
    546 001664  017701  000000G                            MOV     @DECPT,R1                ;; point to DECnet homeblock    ; RJK10
    547 001670  016164  000030  000114                     MOV     D$NLN(R1),L$COU(R4)      ;; get number of channels..
    548 001676  012764  140000  000110                     MOV     #140000,L$ADJ(R4)        ;; scan channel-indexed adjacencies
    549 001704                                             CALL    GETADJ                   ;; get adjacent node
    550 001710  103005                                     BCC     7$                       ;; if CC, found info
    551 001712                                             RETC    R0                       ;; else no active adj. info
    552 001724  116466  000070  000000G 7$:                MOVB    L$LTM(R4),R$R1(SP)       ;; return current listen timer...
    553 001732  016467  000112  176040                     MOV     L$NOD(R4),NODE           ;; ...and node address
    554 001740                                             RETURN                           ;; back to user state at 60$
    555                                             ;
    556                                             ; Ethernet circuits (Router): initialize adjacency scan
    557                                             ;
    558 001742  032764  100000  000064  10$:               BIT     #LF$SEG,L$FLG(R4)        ;; preceding partial response ?
    559 001750  001021                                     BNE     20$                      ;; if NE, yes - already initialized
    560 001752  017701  000000G                            MOV     @DECPT,R1                ;; initialize count:            ; RJK10
    561 001756  016164  000054  000114                     MOV     D$NBRA(R1),L$COU(R4)     ;; broadcast router adjacencies.
    562 001764  066164  000056  000114                     ADD     D$NBEA(R1),L$COU(R4)     ;; plus broadcast endnode adjacencies
    563 001772  066164  000030  000114                     ADD     D$NLN(R1),L$COU(R4)      ;; plus number of channels
    564 002000  012764  140000  000110                     MOV     #140000,L$ADJ(R4)        ;; save start address of adj scan
    565 002006                                             CALL    GETADJ                   ;; get the first adjacency
    566 002012  103420                                     BCS     30$                      ;; if CS, no information
    567                                             ;
    568                                             ; Router: get listen timer for next adjacent node
    569                                             ;
    570 002014  116466  000070  000000G 20$:               MOVB    L$LTM(R4),R$R1(SP)       ;; return current listen timer...
    571 002022  016467  000112  175750                     MOV     L$NOD(R4),NODE           ;; ...and node address
    572 002030                                             CALL    GETADJ                   ;; get next adjacency parameters
    573 002034  103414                                     BCS     40$                      ;; if CS, last adjacency
    574 002036  052764  100000  000064                     BIS     #LF$SEG,L$FLG(R4)        ;; indicate partial response
    575 002044  112774  000003  000060                     MOVB    #MS.PAR,@L$BUF(R4)       ;; change return code to partial
    576 002052  000413                                     BR      55$                      ;; success
    577
    578 002054                          30$:               RETC    R0                       ;; failure
    579 002066  042764  100000  000064  40$:               BIC     #LF$SEG,L$FLG(R4)        ;; end partial response sequence
    580 002074  112774  000001  000060                     MOVB    #MS.SUC,@L$BUF(R4)       ;; ...
    581
    582 002102                          55$:               RETURN                           ;; back to user state at 60$
    583
    584 002104  103417                  60$:               BCS     90$                      ; if CS, no info
    585 002106  105701                                     TSTB    R1                       ; check format of returned data
    586 002110  100406                                     BMI     70$                      ; if MI, very high value
    587 002112  042701  177400                             BIC     #177400,R1               ; clear out any leading junk
    588 002116                                             ASL$    3,R1                     ; else its seconds * 8
    589 002124  000404                                     BR      80$                      ; done
    590 002126  042701  177600          70$:               BIC     #^C177,R1                ; isolate meaningful bits
    591 002132  000301                                     SWAB    R1                       ; get seconds * 256...
    592 002134  006301                                     ASL     R1                       ; ....seconds * 512
    593 002136  010167  175652          80$:               MOV     R1,LTIM                  ; pass back result
    594 002142  000241                                     CLC                              ; success
    595
    596 002144                          90$:               RESRG   <R2,R1>                  ; restore registers
    597 002150                                             RETURN
    598
```

```
                                                                 G 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-5
Symbol table

R.STBL  000210           S$STA   000003           S.RBE   000042           TR.TLZ  000072           VF$RCV= 100000
SETNAM  003320R          S$TIM   000014           S.RCB   000020           TX.NXM= 000002           VF$XMT= 040000
SETNOD  002266R          S$TMR   000016           S.RCV   000014           TX.OVR= 000001           V$FLG   000000
SETPAR= ****** GX        S$WND   000006           S.RCVB  000072           TX.TMO= 000004           V$LEN   000022
SETSER  003234R          S$$WRG= 000000           S.RCVC  000056           T$FLAG  000044           V$RCV   000002
SE.IRS= 004000           S$$YSZ= 007600           S.RDE   000034           T$LIF   000013           V$XMT   000012
SE.NRS= 002000           S.ALF   000044           S.REPR  000050           T$LIFL  000013           V$$CTR= 001000
SE.RBS= 001000           S.BLKC  000033           S.REPS  000047           T$LIFO  000013           X$$DBT= 000000
SE.RBU= 000400           S.COST  000001           S.RST   000036           T$LIFS  000013           X$$TLC= 000001
SE.RCH= 000001           S.COUB  000044           S.RTEC  000042           T$LIN   000000           ZF.COU= 001000
SE.RDC= 000002           S.COUL= 000040           S.RTY   000001           T$LIPS  000006           ZF.DDM= 000001
SE.RRR= 000004           S.CTL   000006           S.SELC  000066           T$LLD   000012           ZF.DIA= 004000
SE.SBS= 000200           S.DLCF  000102           S.SELT  000053           T$LLDC  000045           ZF.DLC= 000002
SE.SBU= 000100           S.ERR   000010           S.STA   000000           T$LLDL  000012           ZF.DVP= 100000
SE.SDC= 000020           S.FLG   000000           S.STE   000002           T$LLDO  000012           ZF.INI= 040000
SE.SHC= 000010           S.FNCP  000004           S.STEC  000043           T$LLDS  000012           ZF.KMX= 000020
SE.SRR= 000040           S.HTNA  000023           S.STLG  000104           T$LLEN  000046           ZF.LLC= 000004
SF$ANM  000001           S.LBE   000044           S.STLN  000035           T$LOPR  000002           ZF.LMC= 000100
SF$BLK= 000001           S.LEN   000004           S.STPN  000034           T$LTCL  000024           ZF.MAN= 020000
SF$DNM  000002           S.LGR   000031           S.TIMC  000054           T$LTIM  000026           ZF.MFL= 000010
SF$INC= 000002           S.LGTH= 000050           S.TLZ   000012           T$LTPR  000014           ZF.MTM= 000400
SF.ACT= 000200           S.LIN   000004           S.TMO   000040           T$LTPS  000020           ZF.MUX= 000040
SF.ENA= 000100           S.LINK  177776           S.TNRP  000032           T$NAPL  000004           ZF.PSE= 002000
SF.LPB= 000004           S.LMA   000026           S.TTEC  000041           T$NFE   000000           ZF.SLI= 010000
SF.MFL= 000040           S.LMOS  000030           S.WFA   000016           T$NLEN  000010           ZF.TIM= 000200
SF.PAC= 000020           S.LMRT  000027           S.XDE   000036           T$NNUL  000002           ZF.X3P= 000000
SF.REA= 000010           S.LMS   000025           S.XID   000002           T$NOPL  000006           ZS.ASN= 100000
SF.SER= 000001           S.LNK   000000           S.XMB   000030           T$NRNI  000042           ZS.BSY= 140000
SF.SVC= 000002           S.LSA   000005           S.XMP   000024           T$NRPL  000005           Z.AVL   000014
SF.UNL= 000040           S.NCUB  000000           S.XMT   000012           T$NRUL  000007           Z.DAT   000016
SLTMA = ****** GX        S.NCUL= 000036           S.XMTB  000076           T$NVR   000001           Z.DSP   000000
SVCADD  000006R          S.NKLR  000040           S.XMTC  000062           T$RPRI  000040           Z.FLG   000010
SVCBLK  003114R          S.NKR   000037           TF.FFE= 000004           T$SVC   000034           Z.LEN = 000016
SVCMAP  000004R          S.NKRB  000052           TF.ROF= 000001           T$T5    000030           Z.LLN   000006
SVCRCL  003006R          S.NKRW  000045           TF.RTO= 000002           T$T6    000032           Z.MAP   000020
S$ADSZ  000007           S.NKSB  000051           TF.TAE= 000010           T$$KMG= 000000           Z.NAM   000004
S$CTL   000002           S.NKSW  000046           TF.TOF= 000004           T$$MIN= 000000           Z.PCB   000012
S$DTE   000020           S.NMST  000002           TF.TTO= 000010           T.ACT   000031           Z.SCH   000007
S$FLG   000010           S.NTD   000024           TF.URE= 000001           T.CHA   000032           $C5TA = ****** GX
S$LEN   000032           S.OWNR  000003           TF.USA= 000002           T.CSR   000022           $HEADR= ****** GX
S$LFT   000012           S.PLA   000010           TR.ALF  000102           T.ITM   000001           $$LCNT= 000072 G
S$LNK   000000           S.PLC   000022           TR.DEV  000100           T.QUE   000024           $$LLIN= 000104 G
S$OWNR  000030           S.PLL   000006           TR.LEN= 000210           T.QU2 = 000026           $$$R  = 000010
S$PKSZ  000004           S.PLS   000011           TR.MST  000074           T.RTY   000030           $$$SYS= 004374
S$RCL   000013           S.PSA   000003           TR.PRE  000076           T.TMR   000000           $$$SYX= 000000


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        003356    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 323
Work  file writes: 239
Size of work file: 45431 Words  ( 178 Pages)
```

```
                                                                 H 13
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-6
```

Table of contents

```
     5-    55     MACRO CALLS AND LOCAL DEFINITIONS
     6-    67     CONTEXT AREA DEFINITIONS
     7-    69     BIT DEFINITIONS
     8-    72     LOCAL DATA
     9-    80     DLC LINE COUNTERS DISPATCH TABLE
    10-   115     CIRECO - READ/ZERO LINE COUNTERS
```

```
CIRECO - READ/AND OR ZERO CIRCU MACRO V05.03b Saturday 29-Jun-85 12:17  Page 10-5
Symbol table

ZF.LMC= 000100          ZF.PSE= 002000          ZS.BSY= 140000          Z.LEN = 000016          Z.SCH   000007
ZF.MAN= 020000          ZF.SLI= 010000          Z.AVL   000014          Z.LLN   000006          $PRCCO= ****** GX
ZF.MFL= 000010          ZF.TIM= 000200          Z.DAT   000016          Z.MAP   000020          $$$SYS= 004374
ZF.MTM= 000400          ZF.X3P= 000000          Z.DSP   000000          Z.NAM   000004          $$$SYX= 000000
ZF.MUX= 000040          ZS.ASN= 100000          Z.FLG   000010          Z.PCB   000012

. ABS.  177776     000     (RW,I,GBL,ABS,OVR)
        000136     001     (RW,I,LCL,REL,CON)
$$DLC0  000030     002     (RW,I,LCL,REL,CON)
$$DLC1  000002     003     (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 213
Work  file writes: 147
Size of work file: 30279 Words  ( 119 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:26.29
SY:CIRECO.V2,[135,134]CIRECO/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CIRECO
```

```
Symbol table

MD$FUL  000000     ME.CST  177765     ML$TOP  000001     MP$BRT  001620     MP$EVE  000311
MD$HAL  000001     ME.DIS  177755     MN$UNL  000377     MP$BSA  002176     MP$FNC  001752
MD$HEL  000002     ME.DON  177600     MO$ACC  000200     MP$BSD  002203     MP$GDT  002222
MD$KCP  000013     ME.FCO  177762     MO$ADD  000002     MP$BSI  002200     MP$GNM  002223
MD$KDP  000034     ME.FIO  177756     MO$ALA  000004     MP$BSP  006331     MP$GRO  002115
MD$KDZ  000036     ME.FOP  177763     MO$ALI  000000     MP$BUF  000170     MP$GRP  000541
MD$KL   000040     ME.FOR  177776     MO$CHA  000040     MP$BUP  006324     MP$GTY  002224
MD$KMX  000054     ME.FUN  177777     MO$CIR  000003     MP$BUS  001643     MP$HAD  001757
MD$KMY  000052     ME.GRO  177745     MO$CLE  000100     MP$CAC  001750     MP$HBT  002142
MD$LEN  000050     ME.HAR  177750     MO$COU  000060     MP$CAS  001762     MP$HDD  000162
MD$PCL  000011     ME.IID  177767     MO$DAC  000000     MP$CAT  002210     MP$HTM  001612
MD$QNA  000005     ME.LCO  177766     MO$DEF  000000     MP$CCS  004406     MP$HWA  002210
MD$UNA  000001     ME.LPR  177757     MO$DPR  000000     MP$CHN  002141     MP$IAT  001322
MD$WIT  000002     ME.MPR  177773     MO$ENT  000017     MP$CIR  000144     MP$IDE  000144
MD.ABO  000017     ME.MVE  177771     MO$EVE  000100     MP$CLK  002131     MP$IDP  006327
MD.ACC  000010     ME.OPE  177747     MO$INF  000160     MP$CLN  002126     MP$IHO  000215
MD.AOB  000016     ME.PLO  177751     MO$INS  000002     MP$CLT  002211     MP$INA  002177
MD.BOB  000011     ME.PMI  177743     MO$LIN  000001     MP$CMB  002142     MP$IND  002204
MD.DIA  000007     ME.PNA  177752     MO$LOG  000001     MP$CMK  000537     MP$INI  002201
MD.DOB  000015     ME.PRI  177775     MO$MIR  000031     MP$CMX  002153     MP$ITI  000776
MD.DUM  000002     ME.PTY  177772     MO$NAM  000001     MP$CND  000310     MP$LAA  000231
MD.FAI  000014     ME.PVA  177760     MO$NIC  000023     MP$CNU  001753     MP$LAN  000234
MD.FNA  000001     ME.RES  177761     MO$NOD  000000     MP$COB  000311     MP$LAR  000202
MD.FOB  000006     ME.ROO  177754     MO$OFF  000001     MP$CON  002126     MP$LAS  000012
MD.LOA  000001     ME.SIZ  177774     MO$ON   000000     MP$COS  001604     MP$LBS  000203
MD.LSH  000020     ME.SYS  177746     MO$OPT  000001     MP$CPF  000230     MP$LCO  000226
MD.NOB  000012     ME.UCO  177770     MO$PER  000200     MP$CPL  000232     MP$LCT  000156
MD.NON  177777     MF$ACT  177776     MO$PRO  000002     MP$CPT  000226     MP$LHL  000232
MD.PER  000000     MF$ADD  000000     MO$PRS  000001     MP$CPU  000161     MP$LLE  000227
MD.RES  000004     MF$ADJ  177774     MO$REA  000200     MP$CSZ  001755     MP$LLO  004432
MD.ROB  000005     MF$ALL  177775     MO$SET  000000     MP$CUS  002127     MP$LMB  002152
MD.RSH  000013     MF$BYE  000300     MO$STA  000020     MP$CVA  000540     MP$LMX  002200
MD.SDU  000005     MF$CHA  000023     MO$SUM  000000     MP$DAL  002570     MP$LNA  000144
MD.SLO  000003     MF$DUM  000020     MO$TSK  000000     MP$DCO= 001464     MP$LNO  000233
MD.SNA  000000     MF$EVT  000001     MO$VOL  000000     MP$DDT  002177     MP$LOA  000170
MD.TLO  000004     MF$FUP  000264     MO$ZER  000000     MP$DEL  001131     MP$LOG  004514
MD.UNA  000002     MF$KNO  177777     MP$ACB  000012     MP$DES  000156     MP$LOO  000620
MD.UNR  000003     MF$LOA  000017     MP$ACC  000514     MP$DEV  002114     MP$LTM  001613
MD.UOB  000007     MF$LOO  177775     MP$ACT  004526     MP$DFA  001320     MP$LTY  004533
MD.VOL  000006     MF$REA  000024     MP$ADD  000766     MP$DHO= 001465     MP$LWI  000230
ME$ALI  000010     MF$SIG  177773     MP$ADJ  001440     MP$DIA  000173     MP$MAC  004420
ME$AL2  000003     MF$SPF  000302     MP$ADP  006325     MP$DLB  002571     MP$MAD  001630
ME$ARE  000005     MF$SYS  000026     MP$ADS  004406     MP$DLG  004521     MP$MAP  006323
ME$CIR  000003     MF$TES  000022     MP$ALB  000036     MP$DLI= 001466     MP$MAR  001635
ME$EXA  000000     MF$TRA  000301     MP$AMC  001640     MP$DLT  002200     MP$MAV  006321
ME$EXE  000200     MF$TRI  000021     MP$AMH  001641     MP$DRO  001441     MP$MBN  001636
ME$LIN  000001     MF$ZER  000025     MP$ANB  004420     MP$DST  000454     MP$MBR  001637
ME$LOG  000002     ML$ALL  100000     MP$ASB  000024     MP$DTE  002140     MP$MBU  001642
ME$MOD  000004     ML$CLS  000000     MP$ASC  004432     MP$DTY= 001452     MP$MCB  000156
ME$NOD  000000     ML$CON  000001     MP$AUS  000512     MP$ UA  000207     MP$MCO  001632
ME$NON  177777     ML$EXT  000001     MP$BBT  002165     MP$ UC  000210     MP$MDE  004421
ME$OBJ  000007     ML$FIL  000002     MP$BDF  002164     [illegible]M 000202     MP$MHO  001633
ME$OB2  000004     ML$FIR  000000     MP$BFQ  002121     MP$DUF  002127     MP$MLB  000202
ME$PRO  000005     ML$INT  000000     MP$BLK  001616     MP$DVC  002114     MP$MLK  001306
ME$SYS  000006     ML$KNO  140000     MP$BLO  001452     MP$DWE  001321     MP$MLN  001631
ME.BLO  177744     ML$MON  000003     MP$BMX  002176     MP$EL1  000157     MP$MLP  006333
ME.CON  177753     ML$SYS  000002     MP$BNP  006330     MP$ETY  001605     MP$MRB  002171
```

```
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE           REFERENCES

                         #9-378     9-379    #9-379    #9-380     9-381    #9-381    #9-385     9-386    #9-386
                         #9-387     9-388    #9-388    #9-392     9-393    #9-393    #9-394     9-395    #9-395
                         #9-399     9-400    #9-400    #9-401     9-402    #9-402    #9-406     9-407    #9-407
                         #9-408     9-409    #9-409    #9-413     9-414    #9-414    #9-415     9-416    #9-416
                          9-417    #9-417    #9-422     9-423    #9-423    #9-424     9-425    #9-425    #9-429
                          9-430    #9-430    #9-431     9-432    #9-432    #9-436     9-437    #9-437    #9-438
                          9-439    #9-439     9-440    #9-440     9-441    #9-441    #9-446     9-447    #9-447
                         #9-448     9-449    #9-449    #9-454     9-455    #9-455    #9-456     9-457    #9-457
                         #9-461     9-462    #9-462    #9-463     9-464    #9-464    #9-468     9-469    #9-469
                         #9-470     9-471    #9-471    #9-476     9-477    #9-477    #9-478     9-479    #9-479
                         #9-483    #9-486     9-487    #9-487    #9-491     9-492    #9-492    #9-496     9-497
                         #9-497    #9-501
HTMACT     000570 R      #18-832
INAACT     001304 R      #35-1196
INDACT     001316 R      #36-1214
INIACT     001330 R      #37-1232
I$$AS    = ******         5-65      27-1046
LCTACT     000212 R      #12-630
LC$NTL   = 000200        #6-72
LC$OWN   = 000400        #6-72
LF$MLT   = 040000        #6-72
LF$REA   = 000001        #6-72
LF$SEG   = 100000        #6-72
LF$SKP   = 000004        #6-72
LF$VR2   = 000010        #6-72
LF$ZER   = 000002        #6-72
LF.ACT   = 100000        #5-67
LF.BRO   = 000400        #5-67
LF.BWT   = 000007        #5-67
LF.ENA   = 002000        #5-67
LF.LPB   = 001000        #5-67
LF.MDC   = 000100        #5-67
LF.MFL   = 004000        #5-67
LF.MTP   = 000020        #5-67
LF.PAC   = 000200        #5-67
LF.RDY   = 040000        #5-67
LF.REA   = 010000        #5-67
LF.SER   = 000040        #5-67
LF.TIM   = 000010        #5-67
LF.UNL   = 020000        #5-67
LF.X2P   = 000000        #5-67
LN.CLO   = 000000        #5-67
LN.DUM   = 000005        #5-67
LN.LOA   = 000004        #5-67
LN.LOO   = 000003        #5-67
LN.OAU   = 000003        #5-67
LN.OFF   = 000001        #5-67
LN.ON    = 000000        #5-67
LN.OOP   = 000004        #5-67
LN.OPE   = 000001        #5-67
LN.REF   = 000002        #5-67
LN.SER   = 000002        #5-67
```

```
          .TITLE  CICHOP - Circuit Change Parameters Operate (Main routine)
          .IDENT  /V05.00/
          .ENABL  LC
          .NLIST  BEX
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CIRCUIT CHANGE PARAMETERS OPERATE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  26-JUN-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;


;
; 4.02  17-OCT-84               RJK
;       Change Symbols for Vectoring Comm Exec
;
```

```
                                                 .SBTTL  FNDCTR -  FIND COUNTER BLOCK
                                         ;+
                                         ;
                                         ; FNDCTR - FIND TRANSPORT LINE COUNTER BLOCK
                                         ;
                                         ; INPUTS:
                                         ;        R4 - ADDRESS OF CONTEXT AREA
                                         ;
                                         ; OUTPUTS:
                                         ;        CARRY CLEAR:
                                         ;           R1 - ADDRESS OF FOUND COUNTER BLOCK
                                         ;        CARRY SET:
                                         ;           COUNTER BLOCK NOT FOUND
                                         ;
                                         ; ***    NOTE: THIS ROUTINE MUST BE CALLED FROM SYSTEM STATE.
                                         ;               APR6 IS MODIFIED
                                         ;
                                         ;
                                         ;-
000622                                   FNDCTR::
000622  016400  000042                           MOV     L$SLT(R4),R0             ;; GET THE SLT ADDRESS
000626  116001  000021                           MOVB    L.OWNR(R0),R1            ;; ASSUME OWNER = LINE OWNER
000632  105760  000014                           TSTB    L.NSTA(R0)               ;; Multipoint circuit ?
000636  001404                                   BEQ     10$                      ;; IF EQ, NO
000640  016401  000050                           MOV     L$TPT(R4),R1             ;; GET THE TRIB FLAGS POINTER
000644  116101  000003                           MOVB    S.OWNR(R1),R1            ;; ELSE, GET TRIB OWNER
000650  067701  000000G                  10$:    ADD     @PDVTA,R1                ;; POINT TO PDV TABLE ENTRY     ; RJK02
000654  011101                                   MOV     (R1),R1                  ;; GET THE PDV ADDRESS          ;**-1
000656  022761  114224  000004                   CMP     #^RXPT,Z.NAM(R1)         ;; IS THE OWNER TRANSPORT ?
000664  001045                                   BNE     40$                      ;; IF NE, NO-PARAMETER NOT APPLICABLE
000666  005046                                   CLR     -(SP)                    ;; GET SLN
000670  017705  000000G                          MOV     @SLTMA,R5                ;; POINT TO SLT VECTOR TABLE    ; RJK02
000674  020025                           15$:    CMP     R0,(R5)+                 ;; IS THIS THE CORRECT SLT?     ;**-1
000676  001402                                   BEQ     17$                      ;; BR IF YES
000700  005216                                   INC     (SP)                     ;; ELSE UPDATE SLN
000702  000774                                   BR      15$                      ;; TRY AGAIN
000704  116466  000040  000001           17$:    MOVB    L$TRB(R4),1(SP)          ;; SET TRIB NUMBER ONTO HIGH BYTE
000712  012605                                   MOV     (SP)+,R5                 ;; RETRIEVE SLN/STA
000714                                           MAP     Z.DSP(R1)                ;; MAP TO NSP'S PROCESS SPACE
000722  016101  000016                           MOV     Z.DAT(R1),R1             ;; POINT TO NSP'S DDB
000726  016100  000100                           MOV     N$TLC(R1),R0             ;; GET THE NUMBER OF LINE COUNTERS
                                                 .IF NDF X$$TLC
                                                 MOV     N$TLC+2(R1),R1           ;; POINT TO THE FIRST
                                                 .IFF
000732                                           MAP     N$TLC+2(R1)              ;; MAP TO IT
000740  016101  000104                           MOV     N$TLC+4(R1),R1           ;; POINT TO THE FIRST
                                                 .ENDC
000744                                           BIAS    R1                       ;; SET FOR APR6 BIAS
000754  005300                           20$:    DEC     R0                       ;; AT END OF COUNTERS ?
000756  002410                                   BLT     40$                      ;; IF LT, YES-PARAMETER NOT APPLICABLE
000760  026105  000000                           CMP     T$LIN(R1),R5             ;; ARE THESE THE COUNTERS ?
000764  001403                                   BEQ     30$                      ;; IF EQ, YES - READ THEM
000766  062701  000046                           ADD     #T$LLEN,R1               ;; ELSE, POINT TO NEXT COUNTERS
000772  000770                                   BR      20$                      ;; AND KEEP LOOKING
000774  000241                           30$:    CLC                              ;; INDICATE FOUND IT
000776  000401                                   BR      50$                      ;;
001000  000261                           40$:    SEC                              ;; INDICATE NOT FOUND
001002                                   50$:    RETURN                           ;;
```

```
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13      PAGE 7
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

T$LIFO    000013           #6-64
T$LIFS    000013           #6-64
T$LIN     000000           #6-64       13-421
T$LIPS    000006           #6-64
T$LLD     000012           #6-64
T$LLDC    000045           #6-64
T$LLDL    000012           #6-64
T$LLDO    000012           #6-64
T$LLDS    000012           #6-64
T$LLEN    000046           #6-64       13-423
T$LOPR    000002           #6-64
T$LTCL    000024           #6-64
T$LTIM    000026           #6-64
T$LTPR    000014           #6-64
T$LTPS    000020           #6-64
T$NAPL    000004           #6-64
T$NFE     000000           #6-64
T$NLEN    000010           #6-64
T$NNUL    000002           #6-64
T$NOPL    000006           #6-64
T$NRNI    000042           #6-64
T$NRPL    000005           #6-64
T$NRUL    000007           #6-64
T$NVR     000001           #6-64
T$RPRI    000040           #6-64
T$SVC     000034           #6-64
T$T5      000030           #6-64
T$T6      000032           #6-64
X$$MCB  = ******            6-67       6-67
X$$TLC  = 000001           #4-2        13-412
ZF.COU  = 001000           #6-67
ZF.DDM  = 000001           #6-67
ZF.DIA  = 004000           #6-67
ZF.DLC  = 000002           #6-67
ZF.DVP  = 100000           #6-67
ZF.INI  = 040000           #6-67
ZF.KMX  = 000020           #6-67
ZF.LLC  = 000004           #6-67
ZF.LMC  = 000100           #6-67
ZF.MAN  = 020000           #6-67
ZF.MFL  = 000010           #6-67
ZF.MTM  = 000400           #6-67
ZF.MUX  = 000040           #6-67
ZF.PSE  = 002000           #6-67
ZF.SLI  = 010000           #6-67
ZF.TIM  = 000200           #6-67
ZF.X3P  = 000000           #6-67
ZS.ASN  = 100000           #6-67
ZS.BSY  = 140000           #6-67
Z.AVL     000014           #6-67
Z.DAT     000016           #6-67       13-410
Z.DSP     000000           #6-67       6-67          13-409
```

```
                                                .SBTTL  SETNTL - DISPATCH NETLOADER SET FUNCTIONS
                                        ;+
                                        ; **-SETNTL-DISPATCH NTL SET FUNCTIONS
                                        ;
                                        ; THIS ROUTINE SENDS A REQUEST TO NTL TO LOAD A LINE.
                                        ;
                                        ; INPUTS:
                                        ;       R4 = POINTER TO CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CC, SUCCESS, ELSE,
                                        ;       ERROR CODE IN R0
                                        ;
                                        ; REGISTERS:
                                        ;       R4, R5 ARE PRESERVED
                                        ;-

 000322                                 SETNTL::
 000322 132764  000200  000041                  BITB    #LP$WTR,L$PFG(R4)       ; IS THIS PART OF A WILD TRIB OPERATION ?
 000330 001403                                  BEQ     10$                     ; IF EQ, NO
 000332 105764  000040                          TSTB    L$TRB(R4)               ; ELSE YES, IS THIS THE FIRST TRIB ?
 000336 001023                                  BNE     30$                     ; IF NE, NO - IGNORE THE REQUEST
 000340 032764  000400  000134  10$:    BIT     #LS.OPT,L$NTL+LR.STS(R4); ANY NTL OPTIONS SPECIFIED ?
 000346 001415                                  BEQ     20$                     ; IF EQ, NO - LOAD IT
 000350 016400  000042                          MOV     L$SLT(R4),R0            ; GET THE SLT ADDRESS
 000354 032760  040000  000000                  BIT     #LF.RDY,L.FLG(R0)       ; IS THE LINE ALREADY LOADED ?
 000362 001407                                  BEQ     20$                     ; IF EQ, NO - LOAD IT
 000364 012700  177765                          MOV     #ME.CST,R0              ; ELSE, LINE IS IN WRONG STATE
 000370 012764  000003  000072                  MOV     #ME$CIR,L$PAR(R4)       ; ...
 000376 000261                                  SEC                             ; ...
 000400 000402                                  BR      30$                     ; RETURN WITH ERROR
 000402                                 20$:    CALL    NTLLIN                  ; ELSE, PERFORM NTL SET LINE FUNCTION
 000406                                 30$:    RETURN
```

```
                                        .SBTTL  STAOFF - REQUEST CIRCUIT STATE ''OFF''
                                ;+
                                ; **-STAOFF-REQUEST CIRCUIT STATE TO ''OFF''
                                ;
                                ; THIS ROUTINE IS CALLED TO CHECK THE CURRENT CIRCUIT STATE, AND
                                ; IF LEGAL, CHANGE IT TO OFF.
                                ;
                                ; INPUTS:
                                ;       R2 = ADDRESS OF STATE/OWNER WORD IN SLT
                                ;       R4 = ADDRESS OF CONTEXT BLOCK
                                ;
                                ; OUTPUTS:
                                ;       IF CC, SUCCESS
                                ;       ELSE, R0 CONTAINS ERROR CODE
                                ;-

002100                          STAOFF:
                                ;
                                ;  TELL DLX TO TURN THE CIRCUIT OFF
                                ;
002100  012700  017010                  MOV     #<36*400!10>,R0         ; SET OFF FUNCTION CODE
002104                                  CALL    DLXQIO                  ; ISSUE QIO
002110  012700  177747                  MOV     #ME.OPE,R0              ; ASSUME OPERATION FAILED
002114                                  RETURN                          ; RETURN WITH C-STATUS

                                        .ENDC   ; NDF R$$11S
```

```
Symbol table

A$$CHK= 000000        CS.CMP= 000200        DLXQIO= ****** GX       D$$L11= 000001        FS.RST= 000000
A$$CPS= 000000        CS.DCR= 000400        DMPSET  000424R     002 D$$YNC= 000000        FS.RTN= 001000
A$$PRI= 000000        CS.DEF= 000004        DV.CCL= 000002          D$$YNM= 000000        FS.SET= 005000
A$$TRP= 000000        CS.DEV= 000002        DV.COM= 020000          EXRQN = ****** GX     FS.SFC= 005000
BBTTBL  000120R       CS.DIS= 000040        DV.DIR= 000010          E$NBR   000014        FS.SFR= 006000
BSATBL  000156R       CS.ENA= 000001        DV.EXT= 000400          E$NBS   000020        FS.SFS= 004000
BSDTBL  000170R       CS.ENB= 000020        DV.F11= 040000          E$NCR   000034        FS.SPW= 040000
BSITBL  000202R       CS.ERR= 100000        DV.ISP= 002000          E$NCS   000036        FS.STM= 000000
CB.CCB= 000002        CS.FTL= 001000        DV.MBC= 000400          E$NIC   000044        FS.STP= 002000
CB.DDM= 000040        CS.HCR= 000001        DV.MNT= 100000          E$NLEN  000050        FS.STR= 001000
CB.DLC= 000020        CS.HFE= 002000        DV.MSD= 000100          E$NLLA  000012        FS.TRM= 003000
CB.RDB= 000004        CS.LST= 040000        DV.OSP= 004000          E$NLNK  000000        FS.WLB= 001000
CB.SDB= 000010        CS.MTL= 004000        DV.PSE= 010000          E$NML   000040        FS.XKL= 002000
CB.SLI= 000100        CS.RNG= 000010        DV.REC= 000001          E$NMR   000024        FS.XOF= 010000
CB.XLB= 000001        CS.ROV= 000004        DV.SDI= 000020          E$NMS   000030        FS.XON= 007000
CCBGT = ****** GX     CS.RSN= 010000        DV.SQD= 000040          E$NNOD  000002        FS.ZER= 002000
CC.LLC= 000200        CS.SHU= 000001        DV.SWL= 001000          E$NRT   000042        F$$LVL= 000001
CE.ABO= 100362        CS.SID= 000002        DV.TTY= 000004          E$NRTP  000005        G$$TPP= 000000
CE.DAO= 100346        CS.STR= 000004        DV.UMD= 000200          E$NSEG  000010        G$$TSS= 000000
CE.DIS= 100366        CS.SUC= 000001        D$AMXC  000072          E$NTIM  000046        G$$TTK= 000000
CE.ERR= 100370        CS.TMO= 020000        D$AMXH  000074          E$NUSE  000004        G$$WRD= 000000
CE.ILN= 100350        CS.XUR= 000004        D$ANN   000000          E$STRT  000006        HEADR = ****** GX
CE.LTO= 100356        C$$CKP= 000000        D$BRPR  000102          E$$XPR= 000000        H.CSP   000000
CE.MOP= 100372        C$$ORE= 000400        D$BRTM  000100          FC.CCP= 000020        H.CUIC  000010
CE.NTE= 100361        C$$RSH= 177564        D$DELF  000045          FC.CTL= 000006        H.DSW   000046
CE.RTE= 100376        C.ADD   000034        D$DELW  000046          FC.KCP= 000016        H.DUIC  000012
CE.SRC= 100364        C.BID   000003        D$END = 000104          FC.KIL= 000004        H.EFLM  000004
CE.STP= 100352        C.BUF   000014        D$FNB   000034          FC.MAN= 000024        H.EFSV  000040
CE.TME= 100354        C.BUF1  000014        D$HIOR  000024          FC.MLD= 000026        H.FCS   000050
CE.TMO= 100374        C.BUF2  000024        D$HOST  000022          FC.PCT= 000030        H.FORT  000052
CE.UNS= 100344        C.CNT   000020        D$INAC  000044          FC.PWR= 000022        H.FPSA  000042
CF.CHN= 000001        C.CNT1  000020        D$INCT  000042          FC.RCE= 000002        H.FPSL= 000062
CF.EOM= 000004        C.CNT2  000030        D$IPL   000051          FC.RCP= 000014        H.FPVA  000034
CF.HDR= 000020        C.FLG   000022        D$LID   000020          FC.TIM= 000010        H.GARD  000072
CF.LB = 100000        C.FLG1  000022        D$LNAM  000006          FC.XCP= 000012        H.HDLN  000002
CF.LIN= 000002        C.FLG2  000032        D$LNUM  000014          FC.XME= 000000        H.IPC   000016
CF.SOM= 000010        C.FNC   000010        D$LST   000047          FNDCTR= ****** GX     H.IPS   000014
CF.SYN= 000040        C.LIN   000006        D$MAXC  000064          FS.AST= 000000        H.ISP   000020
CF.TRN= 000100        C.LNK   000000        D$MAXH  000066          FS.CIB= 002000        H.LUN   000076
CLENTL  000410RG      C.MOD   000011        D$MAXV  000070          FS.CRA= 001000        H.NLUN  000074
CM.CIR= 000002        C.NSP   000004        D$MLL   000040          FS.DIS= 013000        H.NML   000061
CM.FMT= 100000        C.PRO   000042        D$MNOD  000041          FS.DVC= 001000        H.ODVA  000022
CM.HRD= 000002        C.RSV   000002        D$NA    000062          FS.ENB= 012000        H.ODVL  000024
CM.LIN= 000000        C.STA   000007        D$NBEA  000056          FS.EXI= 001000        H.OVLY  000054
CM.LOO= 000001        C.STS   000012        D$NBRA  000054          FS.GET= 006000        H.PFVA  000032
CM.XLO= 000004        C.URM   177776        D$NEND= 000054          FS.HLT= 000000        H.RCVA  000036
CP.DCF= 000040        C.XACP  000004        D$NLN   000030          FS.INI= 000000        H.RRVA  000062
CP.HDL= 000007        C.XID   000035        D$NN    000060          FS.KIL= 000000        H.SPRI  000060
CP.PS = 177400        C.XLEN  000044        D$OUTT  000043          FS.LCL= 100000        H.TKVA  000026
CP.PSI= 000200        C.XPLI  000040        D$RETF  000050          FS.LTM= 001000        H.TKVL  000030
CP.XCF= 000100        C.XPT   000034        D$RNN   000002          FS.MNT= 004000        H.VEXT  000056
CP.2FR= 000030        C.XSVC  000042        D$RTMR  000076          FS.MSN= 014000        H.WND   000044
CS.ABO= 000100        C.XTC   000037        D$SEG   000036          FS.REA= 001000        H.X25   000064
CS.BRO= 000002        C.X25   000036        D$SER   000032          FS.RET= 000000        INATBL  000214R
CS.BUF= 000200        DEALUN= ****** GX     D$SQRL  000052          FS.REZ= 003000        INDTBL  000226R
CS.CES= 000002        DECPT = ****** GX     D$$BUG= 177514          FS.RLB= 002000        INITBL  000240R
CS.CHN= 000010        DLXLUN= ****** GX     D$$ISK= 000000          FS.RNG= 011000        INSMSG  000000R
```

```
                                                                        H  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 7
SYMBOL CROSS REFERENCE                                   CREF    04.00
SYMBOL  VALUE           REFERENCES

MP$XMT    002166          8-132
MS$OFF    000001          13-351    13-360    13-374    14-531    15-610    17-700
MS$ON     000000          13-419    13-488    14-529    14-540    14-570    15-612    17-696
MS$SER    000002          13-492    14-533    15-608    15-639
MS.SUC    000001          20-871
MXBTBL    000310 R       #8-142     22-1054
NF.END  = 000200          13-416
NMTTBL    000132 R       #8-131     22-1021
NODMSG    000074 R       #7-101     13-457
NTLLIN    000446 R        9-178     10-205   #11-226
NTLMSG    000035 R       #7-99      11-254
NTLSND  = ******   GX     11-239
N$PLD     000016          20-907    20-908
N$RT1     000000          20-920
N$RT2     000006          20-923
N$$VCT  = ******          12-298    12-301    12-304    18-751    19-802
PDVID   = ******   GX     20-895
PDVTA   = ******   GX     13-382    14-554    20-901    21-990
PR$LCC  = 000010          20-918    20-919
P$CHN     000004          20-911
P$STA1    000022         *20-918
P$STA2    000023         *20-919
R$$MPL  = ******          13-391
R$$PRO  = ******          5-60      6-86      6-86      7-91      7-109     11-242    11-252    13-455    21-1008
R$$RTR  = ******          7-107     21-1008
R$$11D  = ******          5-72
R$$11M  = 000000          5-72
R$$11S  = ******          5-72      7-97      7-107     11-265    13-335    13-464    13-474    13-484    13-503
                          19-831    21-948
SETBBT    000222 RG      #22-1018
SETBSA    000260 RG      #22-1027
SETBSD    000272 RG      #22-1030
SETBSI    000304 RG      #22-1033
SETCOS    002604 RG      #20-853
SETHTM    000566 RG      #12-283
SETINA    000316 RG      #22-1036
SETIND    000330 RG      #22-1039
SETINI    000342 RG      #22-1042
SETMXB    000412 RG      #22-1054
SETNMT    000234 RG      #22-1021
SETNTL    000322 RG      #9-164
SETOWN    000000 RG      #21-954
SETSER    000566 RG      #24-1119
SETSTA    000734 RG      #13-334
SETTH1    000354 RG      #22-1045
SETTH2    000366 RG      #22-1048
SETTH3    000400 RG      #22-1051
SETXMT    000246 RG      #22-1024
SF.ACT  = 000200         #5-73      13-372    13-449    18-765    19-813    19-820    20-883    21-983
SF.ENA  = 000100         #5-73
SF.LPB  = 000004         #5-73
SF.MFL  = 000040         #5-73
```

```
                                                                        I  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 8
```

```
                                                .SBTTL  CICHPX - Set X.25 circuit parameters
                                        ;+
                                        ; *** CICHPX - Set X.25 circuit parameters
                                        ;
                                        ; INPUTS
                                        ;       R4 - Pointer to context area
                                        ;
                                        ; OUTPUTS
                                        ;       R0 - Error code
                                        ;-

000200                                  CICHPX::
000200  012700  000002                          MOV     #MS.MOR,R0              ; Assume successful completion

000204  026427  000050  177777                  CMP     L$TPT(R4),#-1           ; Is this a named PVC?
000212  001002                                  BNE     10$                     ; Branch if not
000214                                          CALLR   SETPVC                  ; Set PVC parameters

000220                                  10$:    CALL    CKDSVC                  ; Is this a DLM SVC?
000224  103416                                  BCS     90$                     ; Branch if not

000226  032764  000004  000076                  BIT     #LS$CUS,L$FLX(R4)       ; Was USAGE specified?
000234  001404                                  BEQ     20$                     ; Branch if not
000236  022764  000000  000260                  CMP     #MU$PER,L$CUS(R4)       ; Cannot allow USAGE PERMANENT yet
000244  001422                                  BEQ     95$

000246  032764  001000  000076          20$:    BIT     #LS$BLK,L$FLX(R4)       ; Was BLOCKING specified
000254  001010                                  BNE     92$                     ; Error if so - not implemented in XPT
000256                                          CALLR   SETSVC                  ; Set SVC parameters

000262  012700  177745                  90$:    MOV     #ME.GRO,R0              ; Indicate invalid parameters supplied
000266  012764  177777  000072                  MOV     #-1,L$PAR(R4)           ; No further information available
000274  000413                                  BR      100$

000276  012700  177752                  92$:    MOV     #ME.PNA,R0              ; Indicate parameter does not apply
000302  012764  001616  000072                  MOV     #MP$BLK,L$PAR(R4)       ; Tell the user what we can't do
000310  000405                                  BR      100$

000312  012700  177760                  95$:    MOV     #ME.PVA,R0              ; Indicate cannot cope with PERMANENT
000316  012764  002127  000072                  MOV     #MP$CUS,L$PAR(R4)
000324                                  100$:   RETURN
```

```
                                                                                                     H 10
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 16-5
Symbol table

X$NRBY  000032          X$PRT   000021          X$SS    000004          X$TYP   000015          $HEADR= ****** GX
X$NRPK  000042          X$PS    000010          X$ST    000005          X$USR   000022          $STPAR= ****** GX
X$NRRE  000053          X$RPR   000013          X$TCLZ  000030          X$WAQ   000062          $XLBOV= 000000
X$NTBY  000036          X$RPS   000012          X$TIMC  000001          X$WSZ   000020          $$$R  = 000010
X$NTPK  000046          X$RTRY  000002          X$TIMR  000000          X$$DBT= 000000          $$$SYS= 004374
X$PKSZ  000016          X$RXI   000104          X$TXI   000102          $ADPVC  001222RG        $$$SYX= 000000
X$PR    000011          X$RXQ   000072          X$TXQ   000066          $CEACC= ****** GX

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        001232    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 285
Work  file writes: 195
Size of work file: 39024 Words  ( 153 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:42.08
SY:CICHPX.V2,[135,134]CICHPX/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CICHPX
```

```
                                                                                                     I 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 1
```

```
                                        .SBTTL  BIT DEFINITIONS
                                ;
                                ; PARSE FLAG DEFINITIONS (L$PFG)
                                ;
        000001                          LP$UNT = 1                      ; UNIT NUMBER FOUND
        000002                          LP$TRB = 2                      ; TRIBUTARY NUMBER FOUND
        000004                          LP$MUX = 4                      ; DEVICE IS MUX
        000010                          LP$MPT = 10                     ; LINE IS MULTIPOINT
        000020                          LP$WDV = 20                     ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40                     ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100                    ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200                    ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK
                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                ;
        000001                          LF$REA = 1                      ; READ COUNTERS OPERATION
        000002                          LF$ZER = 2                      ; ZERO COUNTERS OPERATION
        000004                          LF$SKP = 4                      ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
        000010                          LF$VR2 = 10                     ; CONNECTED TO VERSION 2.0 NCP
        000020                          LF$END = 20                     ; EXECUTOR IS AN ENDNODE
        000040                          LF$SIG = 40                     ; SHOWING SIGNIFIGANT LINES/CIRCUITS
        040000                          LF$MLT = 40000                  ; MULTIPLE ADJACENCY FLAG
        100000                          LF$SEG  = 100000                ; SEGMENTED RESPONSE IN PROGRESS
                                ;
                                ; BIT POSITION DEFINITIONS
                                ;
        000001                          BIT0 = 1
        000002                          BIT1 = 2
        000004                          BIT2 = 4
        000010                          BIT3 = 10
        000020                          BIT4 = 20
        000040                          BIT5 = 40
        000100                          BIT6 = 100
        000200                          BIT7 = 200
        000400                          BIT8 = 400
        001000                          BIT9 = 1000
        002000                          BIT10 = 2000
        004000                          BIT11 = 4000
        010000                          BIT12 = 10000
        020000                          BIT13 = 20000
        040000                          BIT14 = 40000
        100000                          BIT15 = 100000
```

```
                                                 .SBTTL  GETADJ - Get an adjacency
                                         ;+
                                         ; ** GETADJ - Get an adjacency
                                         ;
                                         ; Inputs:        R2 = address of XPT DDB
                                         ;                R4 = context area address
                                         ;                L$ADJ(R4) = current position in adjacency database
                                         ;                L$COU(R4) = count of unscanned entries in adjacency database
                                         ;                L$CHN(R4) = channel number for circuit
                                         ;
                                         ; Output:        carry set:      no more adjacent nodes.
                                         ;                carry clear:    success
                                         ;
                                         ;                L$NOD(R4) = adjacent node address
                                         ;                L$LTM(R4) = encoded listener timer
                                         ;
                                         ; Registers:     R1 is corrupted.
                                         ;
                                         ; N.B.           ***  This routine changes APR 6 mapping and   ***
                                         ;                ***  must be called from system state         ***
                                         ;-

    002152                               GETADJ:
    002152                                       SAVMAP                                  ;; save mapping
    002156                                       MAP     N$ADJ1[illegible]2)             ;; map to ADJ1

    002164  005764  000114               10$:    TST     L$COU(R4)                       ;; any unscanned adjacencies ?
    002170  003432                               BLE     20$                             ;; if LE, no
    002172  016401  000110                       MOV     L$ADJ(R4),R1                    ;; else get next adjacency
    002176  062764  000004  000110               ADD     #A$LEN,L$ADJ(R4)                ;; advance pointer
    002204  005364  000114                       DEC     L$COU(R4)                       ;; change unscanned count

    002210  132761  000010  000002               BITB    #AT$UP,A$TYP(R1)                ;; is this adjacency up ?
    002216  001762                               BEQ     10$                             ;; if EQ, no
    002[illegible]  126164  000003  000054       CMPB    A$CIR(R1),L$CHN(R4)             ;; is it for correct channel ?
    002226  001356                               BNE     10$                             ;; if NE, keep walking

    002230  016164  000000  000112               MOV     A$NID(R1),L$NOD(R4)             ;; else get node address
    002236  001752                               BEQ     10$                             ;; br if no entry
    002240                                       MAP     N$ADJ2(R2)                      ;; map to ADJ2 database
    002246  116164  000002  000070               MOVB    A$TMI(R1),L$LTM(R4)             ;; get encoded listen timer
    002254  005727                               TST     (PC)+                           ;; return success
    002256  000261                       20$:    SEC                                     ;; return failure
    002260                                       RESMAP                                  ;; restore mapping
    002264                                       RETURN                                  ;; and return success
```

```
CIRECH - READ CIRCUIT CHARACTER MACRO V05.03b Saturday 29-Jun-85 12:15  Page 21-6
Symbol table

Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:03.24
SY:[IRECH.V2,[135,134]CIRECH/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]V2,CIRECH
```

```
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16     PAGE 1
```

```
    .TITLE  CIRECO - READ/AND OR ZERO CIRCUIT COUNTERS
    .IDENT  /V05.00/
    .ENABL  LC

;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - READ AND ZERO CIRCUIT COUNTERS
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  01-JUL-81
;       DECNET-11M/S V3.1
;       DECNET-11M-PLUS V1.1
;
; 2.00  16-APR-82
;       DECNET-11M V3.1
;       DECNET-11M-PLUS V1.1
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
CIRECO     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE             REFERENCES

BIT0    = 000001          #7-69
BIT1    = 000002          #7-69
BIT10   = 002000          #7-69
BIT11   = 004000          #7-69
BIT12   = 010000          #7-69
BIT13   = 020000          #7-69
BIT14   = 040000          #7-69
BIT15   = 100000          #7-69
BIT2    = 000004          #7-69
BIT3    = 000010          #7-69
BIT4    = 000020          #7-69
BIT5    = 000040          #7-69
BIT6    = 000100          #7-69
BIT7    = 000200          #7-69
BIT8    = 000400          #7-69
BIT9    = 001000          #7-69
B$10      000003          #6-69
B$11      000004          #6-69
B$12      000005          #6-69
B$5       000000          #6-69
B$6       000001          #6-69
B$7       000002          #6-69
B$7S10    000006          #6-69
CIRECO    000002 RG       #10-133
DACIR   = ******   GX       9-99
DCPCIR  = ******   GX       9-95
DLCTBL    000000 R        #9-92        10-156
DLMCOU  = ******   GX       9-102
DMPCIR  = ******   GX       9-100      9-101
I$$AS   = ******            5-64
LF$END  = 000020          #7-69
LF$MLT  = 040000          #7-69
LF$REA  = 000001          #7-69
LF$SEG  = 100000          #7-69
LF$SIG  = 000040          #7-69
LF$SKP  = 000004          #7-69
LF$VR2  = 000010          #7-69
LF$ZER  = 000002          #7-69
LF.ACT  = 100000          #5-63
LF.BRO  = 000400          #5-63
LF.BWT  = 000007          #5-63
LF.ENA  = 002000          #5-63
LF.LPB  = 001000          #5-63
LF.MDC  = 000100          #5-63
LF.MFL  = 004000          #5-63
LF.MTP  = 000020          #5-63
LF.PAC  = 000200          #5-63
LF.RDY  = 040000          #5-63        10-149
LF.REA  = 010000          #5-63
LF.SER  = 000040          #5-63
LF.TIM  = 000010          #5-63
LF.UNL  = 020000          #5-63
```

```
Symbol table

MP$MRO   001605      MP$RET   001631      MP$USR   001750      MT$CON   000001      M$$MGE=  000000
MP$MRP   006322      MP$RFA   001323      MP$VEC   004410      MT$DMC   000004      M$$NET=  000000
MP$MRT   002153      MP$RMX   002201      MP$VER   004406      MT$ETH   000006      M$$OVR=  000000
MP$MRV   006320      MP$RPA   004374      MP$WDF   002165      MT$HEX   000040      NF$ACC=  100000
MP$MSB   000170      MP$RPR   001606      MP$WMX   002177      MT$LPB   000005      NF$EVT=  040000
MP$MVE   000145      MP$RRT   002213      MP$XMT   002166      MT$MAX   000037      NF.ACC=  000001
MP$MVI   001634      MP$RST   002212      MP$XPF   000242      MT$MUL   000100      NF.BCP=  000040
MP$MVR   001751      MP$RSV   001754      MP$XPL   000244      MT$NLE   000017      NF.END=  000200
MP$MWN   002154      MP$RSZ   001756      MP$XPT   000240      MT$NON   000001      NF.EVT=  000002
MP$MXB   002172      MP$RTI   001616      MP$XXX   177777      MT$NR4   000005      NF.LLI=  000004
MP$MXC   000466      MP$RTM   001611      MP$5ST   004406      MT$NTY   000060      NF.LV2=  000100
MP$MXR   001630      MP$RVE   001604      MS$ACT   000001      MT$OCT   000060      NF.SLI=  000400
MP$MXW   002143      MP$RVT   000156      MS$ADU   000010      MT$PHA   000002      NF.SMC=  000020
MP$NAA   006335      MP$SAC   000514      MS$ALO   000007      MT$POI   000000      NF.SWP=  000010
MP$NAC   001130      MP$SAD   000543      MS$ASE   000006      MT$QP2   000010      NM.CLN   003400
MP$NAP   006326      MP$SCA   000310      MS$ATR   000011      MT$ROU   000000      NM.INI   002400
MP$NCT   000240      MP$SCO   000144      MS$AUT   000000      MT$RO4   000004      NM.DPR   003000
MP$NET   002114      MP$SCT   002176      MS$CLE   000003      MT$SEC   000000      NM.VR2   002401
MP$NLI   000765      MP$SDU   000203      MS$DED   000004      MT$SGD   000020      NM.VR3   002402
MP$NNA   000764      MP$SDV   000160      MS$DIE   000003      MT$SYS   000002      N$$ACC=  000001
MP$NOD   000500      MP$SEH   004401      MS$DUM   000004      MT$TER   000001      N$$BUF=  000001
MP$NRA   006334      MP$SER   000144      MS$FAI   000013      MT$TRI   000002      N$$LDV=  000001
MP$NRB   000214      MP$SET   000000      MS$GLO   000000      MT$TYP   007777      N$$MCP=  000001
MP$NSA   001617      MP$SGZ   001644      MS$HOL   000002      MT$USD   000000      N$$MLL=  000001
MP$NTI   002141      MP$SID   000176      MS$INA   000002      MT$X25   000003      N$$MOV=  000010
MP$NUM   001642      MP$SIN   000310      MS$LOA   000003      MU$INC   000001      N$$NCT=  000001
MP$NVE   001274      MP$SLI   000156      MS$LOO   000002      MU$OUT   000002      N$$PEM=  000001
MP$NXN=  001476      MP$SLO   000171      MS$OFF   000001      MU$PER   000000      P$$P45=  000000
MP$OAC   000632      MP$SMX   002202      MS$ON    000000      MV$III   000000      P$$WRD=  000000
MP$OAN   000620      MP$SND   000500      MS$PRO   000006      MV$IV    000001      Q$$OPT=  000010
MP$OCO   000776      MP$SNP   006332      MS$REA   000004      MX$ACT   000020      R$$DER=  000000
MP$OHO   000214      MP$SNU   000542      MS$REF   000001      MX$CIR   000020      R$$K11=  000001
MP$ONA   000764      MP$SOB   000524      MS$RES   000003      MX$CLN   000020      R$$SND=  000000
MP$ONR   002114      MP$SPA   000157      MS$RST   000001      MX$CNM   000006      R$$11M=  000000
MP$OQL   001453      MP$SPR   000536      MS$RSX   000002      MX$CON   000006      SF$ANM   000001
MP$OTI   000777      MP$SPS   000513      MS$RT    000005      MX$DAC   000020      SF$DNM   000002
MP$OTY   001022      MP$STA   000000      MS$SER   000002      MX$DTE   000020      S$$WRG=  000000
MP$OUS   000777      MP$STI   002140      MS$SHU   000002      MX$FIL   000034      S$$YSZ=  007600
MP$OVE   001010      MP$STT   002201      MS$SRV   000007      MX$LGC   000777      S1.DEC=  001000
MP$OWN   004374      MP$STY   000175      MS$STA   000000      MX$LIN   000020      S1.DPR=  000400
MP$PAR   000036      MP$SUB   000001      MS$SYN   000012      MX$LON   000006      S1.DSI=  004000
MP$PAS   000513      MP$SUR   000156      MS$TER   000001      MX$NMS   000454      S1.ESC=  000002
MP$PCH   002152      MP$SUS   000512      MS$TOP   000003      MX$NOD   000006      S1.IBF=  002000
MP$PCO   000024      MP$SVR   000163      MS$TRI   000005      MX$OBJ   000006      S1.IBY=  000200
MP$PCT   000144      MP$SWI   002260      MS$UNR   000005      MX$OWN   000040      S1.OBY=  000100
MP$PDT   002114      MP$SYP   002424      MS$VMS   000004      MX$PAR   000036      S1.PTH=  000010
MP$PHA   000170      MP$TFL   000156      MS.DON   177600      MX$PAS   000010      S1.RES=  010000
MP$PHY   000012      MP$TH1   002206      MS.MOR   000002      MX$RAC   000047      S1.RNE=  000020
MP$PLI   000025      MP$TH2   002205      MS.PAR   000003      MX$RID   000047      S1.RNF=  020000
MP$PLN   002140      MP$TH3   002202      MS.SUC   000001      MX$RPS   000047      S1.RSP=  000004
MP$PLD   000012      MP$TLN   000202      MT$AR4   000003      MX$SID   000040      S1.RST=  000001
MP$PMC   004374      MP$TLO   000172      MT$ASC   000100      MX$SNK   000377      S1.TNE=  040000
MP$PNT   002126      MP$TPA   004375      MT$BIL   000001      MX$TYP   000077      S1.TSY=  000040
MP$PRI   004411      MP$TRI   002164      MT$BYS   000011      MX$UID   000020      S1.USI=  100000
MP$PRO   002130      MP$TST   000144      MT$CI    000007      M$$CRB=  000124      S2.BEL=  020000
MP$PSS   001763      MP$TYP   002130      MT$CLE   000077      M$$CRX=  000000      S2.BRQ=  000020
MP$PST   001762      MP$UCS   004407      MT$COD   000200      M$$FCS=  000000      S2.CR =  002000
```

```
                                                             I  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11     PAGE 3
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL   VALUE              REFERENCES

LN.STA  = 000017            #5-67
LN.SUB  = 000360            #5-67
LN.TRI  = 000006            #5-67
LO.CON    000006            #5-64
LO.CSR    000006            #5-64
LO.CTL    000014            #5-64
LO.LIN    000004            #5-64
LO.PAR    000010            #5-64
LO.PRI    000010            #5-64
LO.SCR    000014            #5-64
LO.STA    000015            #5-64
LO.VCT    000012            #5-64
LP$MPT  = 000010            #6-72
LP$MUX  = 000004            #6-72
LP$NXC  = 100000            #6-72
LP$TRB  = 000002            #6-72          17-811
LP$UNT  = 000001            #6-72
LP$WCN  = 000040            #6-72          6-72
LP$WDV  = 000020            #6-72          6-72
LP$WLD  = 000360            #6-72          14-719
LP$WTR  = 000200            #6-72          6-72
LP$WUN  = 000100            #6-72          6-72
LR.AST    000026            #5-64
LR.CIR    000002            #5-64
LR.CTL    000004            #5-64
LR.EFN    000004            #5-64
LR.LIN    000002            #5-64
LR.PRM    000030            #5-64
LR.PRO    000002            #5-64
LR.STS    000000            #5-64
LR.TCB    000002            #5-64
LR.UNT    000005            #5-64
LS$ACT  = 000040            #6-72
LS$BBT  = 010000            #6-72
LS$BLK  = 001000            #6-72          6-72
LS$BSA  = 000001            #6-72
LS$BSD  = 000002            #6-72
LS$BSI  = 000004            #6-72
LS$CHN  = 000001            #6-72          6-72
LS$CMB  = 000002            #6-72          6-72          6-72          6-72
LS$COS  = 000004            #6-72
LS$CUS  = 000004            #6-72          6-72
LS$DDT  = 001000            #6-72
LS$DEA  = 000100            #6-72
LS$DLM  = 001004            #6-72
LS$DLT  = 002000            #6-72
LS$DTE  = 000010            #6-72          6-72
LS$HBT  = 000400            #6-72
LS$HTM  = 010000            #6-72
LS$INA  = 000010            #6-72
LS$IND  = 000020            #6-72
LS$INI  = 000040            #6-72
```

```
                                                             J  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11     PAGE 4
```

```
                                          .SBTTL  MACRO CALLS AND LOCAL DATA
                                          .MCALL  BIAS,CICCX$,CTRDF$,MANDF$,MAP,NTLDF$
                                          .MCALL  PDVDF$,RESRG,SAVRG,SLTDF$,XPDDB$


   000000                                 CTRDF$                          ; Define transport counters
   000000                                 MANDF$                          ; Define management symbols
   000000                                 NTLDF$                          ; Define NTL symbols
   000000                                 PDVDF$                          ; Define PDV symbols
   000000                                 SLTDF$                          ; Define SLT symbols
   000000                                 XPDDB$                          ; Define XPT database
```

```
    429
    430                                          .ENDC    ; NDF R$$11S
    431
    432         000001                           .END
```

```
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 8
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE              REFERENCES

Z.FLG     000010           #6-67
Z.LEN   = 000016           #6-67
Z.LLN     000006           #6-67
Z.MAP     000020           #6-67
Z.NAM     000004           #6-67          13-399
Z.PCB     000012           #6-67
Z.SCH     000007           #6-67
$ADPVC  = ******   GX        9-180
$SAVRG  = ******   GX        9-152
```

```
CICHOP     CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 9
```

```
                                                .SBTTL  CLENTL - DISPATCH NETLOADER CLEAR FUNCTIONS
                                        ;+
                                        ; **-CLENTL-DISPATCH NTL CLEAR FUNCTIONS
                                        ;
                                        ; THIS ROUTINE SENDS A REQUEST TO NTL TO CLEAR A LINE.
                                        ;
                                        ; INPUTS:
                                        ;       R4 = POINTER TO CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CC, SUCCESS, ELSE,
                                        ;       ERROR CODE IN R0
                                        ;
                                        ; REGISTERS:
                                        ;       R4, R5 ARE PRESERVED
                                        ;-

000410                                  CLENTL::
000410  132764  000010  000041                  BITB    #LP$MPT,L$PFG(R4)       ; MULTIPOINT LINE ?
000416  001410                                  BEQ     10$                     ; IF EQ, NO - CLEAR IT
000420  132764  000200  000041                  BITB    #LP$WTR,L$PFG(R4)       ; IS THIS PART OF A WILD TRIB OPERATION ?
000426  001404                                  BEQ     10$                     ; IF EQ, NO
000430  126427  000052  000001                  CMPB    L$TNM(R4),#1            ; ELSE YES, IS THIS THE LAST TRIB ?
000436  001002                                  BNE     20$                     ; IF NE, NO - SKIP THE REQUEST
000440                                  10$:    CALL    NTLLIN                  ; ELSE, PERFORM NTL CLEAR LINE FUNCTION
000444                                  20$:    RETURN
```

```
                                                .SBTTL  TWKLIN - TWEAK A CIRCUIT'S STATE
                                        ;+
                                        ; **-TWKLIN-TWEAK A CIRCUIT'S STATE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO CHANGE THE STATE OF CIRCUIT WHEN DLX
                                        ; IS NOT IN THE SYSTEM. LEGAL STATES ARE LIMITED TO "ON" AND "OFF".
                                        ;
                                        ; INPUTS:
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;            L$STA(R4) = NEW STATE
                                        ;
                                        ; OUTPUTS:
                                        ;       CIRCUIT TURNED ON OR OFF
                                        ;       ERROR RETURN IN R0
                                        ;
                                        ; REGISTERS:
                                        ;       R4, R5 ARE PRESERVED
                                        ;-

 002116                                 TWKLIN::
 002116 012700  177765                          MOV     #ME.CST,R0              ; ASSUME CIRCUIT IS IN WRONG STATE
 002122 012764  000003  000072                  MOV     #ME$CIR,L$PAR(R4)       ; ...

 002130 122764  000000  000100                  CMPB    #MS$ON,L$STA(R4)        ; TURN CIRCUIT ON ?
 002136 001003                                  BNE     20$                     ; IF NE, NO
 002140                                         CALL    LINON                   ; ELSE TURN IT ON
 002144 000412                                  BR      40$                     ; AND RETURN
 002146 122764  000001  000100          20$:    CMPB    #MS$OFF,L$STA(R4)       ; TURN CIRCUIT OFF ?
 002154 001404                                  BEQ     30$                     ; IF EQ, YES - DO IT
 002156 012700  177747                          MOV     #ME.OPE,R0              ; ELSE, OPERATION FAILED
 002162 000261                                  SEC                             ; ...
 002164 000402                                  BR      40$                     ; ...
 002166                                 30$:    CALL    LINOFF                  ; TURN IT OFF
 002172                                 40$:    RETURN
```

```
                                                                 I 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-2
Symbol table

I$$RAR= 000000         LO.SCR  000014         LS$OWN= 000010         L$FLX   000076         L$$11R= 000000
I$$RDN= 000000         LO.STA  000015         LS$PLT= 004000         L$FL1   000066         L.COST  000015
I.OFF = 000024         LO.VCT  000012         LS$PRO= 010000         L$FL2   000070         L.CTL   000012
KISAR = ****** GX      LP$MPT= 000010         LS$PVC= 020053         L$HBT   000254         L.CVA   177776
K$$CNT= 177546         LP$MUX= 000004         LS$RET= 000200         L$HTM   000226         L.DDM   000002
K$$CSR= 177546         LP$NXC= 100000         LS$SER= 100000         L$INA   000125         L.DDS   000004
K$$LDC= 000000         LP$TRB= 000002         LS$STA= 000002         L$IND   000126         L.DLC   000003
K$$TPS= 000074         LP$UNT= 000001         LS$SVC= 000362         L$INI   000127         L.DLM   000006
LC$EXT  000000         LP$WCN= 000040         LS$TAD= 000020         L$LCT   000224         L.DLS   000010
LC$INT  000001         LP$WDV= 000020         LS$TH1= 000100         L$LEN   000264         L.FLG   000000
LC$NTL= 000200         LP$WLD= 000360         LS$TH2= 000200         L$LMB   000234         L.KRBA  000016
LC$OWN= 000400         LP$WTR= 000200         LS$TH3= 000400         L$LTM   000230         L.LEN = 000022
LD$LP = 000000         LP$WUN= 000100         LS$XMT= 040000         L$MRT   000253         L.MPF   000022
LF$MLT= 040000         LR.AST  000026         LS.AST= 100000         L$MSG   000056         L.NMST  000020
LF$REA= 000001         LR.CIR  000002         LS.CEX= 000001         L$MTYP  000075         L.NSTA  000014
LF$SEG= 100000         LR.CTL  000004         LS.CIR= 000010         L$MWN   000252         L.OWNR  000021
LF$SKP= 000004         LR.EFN  000004         LS.CXO= 010000         L$MXB   000133         L.UNT   000013
LF$VR2= 000010         LR.LIN  000002         LS.CX5= 000400         L$MXR   000253         MB$DIS  000001
LF$ZER= 000002         LR.PRM  000030         LS.DLM= 004000         L$MXW   000252         MB$ENA  000000
LF.ACT= 100000         LR.PRO  000002         LS.ECH= 001000         L$NAM   000000         MB$FUL  000002
LF.BRO= 000400         LR.STS  000000         LS.FDX= 004000         L$NLEN= 000020         MB$MIX  000002
LF.BWT= 000007         LR.TCB  000002         LS.HDX= 010000         L$NMT   000116         MB$ONE  000001
LF.ENA= 002000         LR.UNT  000005         LS.LIN= 000003         L$NTI   000256         MB$RXO  000001
LF.LPB= 001000         LS$ACT= 000040         LS.LMC= 000007         L$NTL   000134         MB$TXO  000000
LF.MDC= 000100         LS$BBT= 010000         LS.NTI= 004000         L$NUM   000240         MB$ZER  000000
LF.MFL= 004000         LS$BLK= 001000         LS.ON = 000011         L$NUML  000236         MC$BAB  006073
LF.MTP= 000020         LS$BSA= 000001         LS.OPT= 000400         L$NXT   000044         MC$BID  001765
LF.PAC= 000200         LS$BSD= 000002         LS.PRO= 000002         L$OPT   000062         MC$BMC  001767
LF.RDY= 040000         LS$BSI= 000004         LS.PWF= 040000         L$OWN   000102         MC$BSC  001766
LF.REA= 010000         LS$CHN= 000001         LS.TOP= 002000         L$PAR   000072         MC$CAC  001442
LF.SER= 000040         LS$CMB= 000002         LS.UNF= 020000         L$PDV   000055         MC$CAP  001440
LF.TIM= 000010         LS$COS= 000004         LS.X25= 000012         L$PFG   000041         MC$CCL  001445
LF.UNL= 020000         LS$CUS= 000004         LS.11D= 004000         L$PLT   000112         MC$CDC  002045
LF.X2P= 000000         LS$DDT= 001000         LX.CEX= 000004         L$PRO   000075         MC$CDP  001441
LINOFF  002414R        LS$DEA= 000100         LX.LIN= 000006         L$PVC   000004         MC$CIF  001465
LINON   002174R        LS$DLM= 001004         LX.PRO= 000005         L$RET   000256         MC$CLD  001464
LLCRS = ****** GX      LS$DLT= 002000         L$ACT   000104         L$SCN   000020         MC$COU  100000
LN.CLO= 000000         LS$DTE= 000010         L$BBT   000114         L$SCR   000166         MC$CTL  001454
LN.DUM= 000005         LS$HBT= 000400         L$BLK   000262         L$SER   000230         MC$CTR  001452
LN.LOA= 000004         LS$HTM= 010000         L$BSA   000122         L$SLEN  000074         MC$CTS  001453
LN.LOO= 000003         LS$INA= 000010         L$BSD   000123         L$SLT   000042         MC$DOV  002050
LN.OAU= 000003         LS$IND= 000020         L$BSI   000124         L$SNM   000046         MC$LBR  001750
LN.OFF= 000001         LS$INI= 000040         L$BUF   000060         L$STA   000100         MC$LBS  001751
LN.ON = 000000         LS$LCT= 020000         L$CHN   000054         L$TAD   000103         MC$LDI  001774
LN.OOP= 000004         LS$LMB= 000002         L$CMB   000234         L$TH1   000130         MC$LDO  001775
LN.OPE= 000001         LS$LOO= 040000         L$COS   000101         L$TH2   000131         MC$LDR  001762
LN.REF= 000002         LS$MDT= 020000         L$CTB   000053         L$TH3   000132         MC$LDS  001763
LN.SER= 000002         LS$MRT= 000020         L$CTL   000002         L$TNM   000052         MC$LLB  002021
LN.STA= 000017         LS$MWN= 000040         L$CUS   000260         L$TPT   000050         MC$LLP  002115
LN.SUB= 000360         LS$MXB= 001000         L$DDM   000000         L$TRB   000040         MC$LLR  002007
LN.TRI= 000006         LS$MXR= 000020         L$DDT   000106         L$TYP   000063         MC$LOO  000001
LO.CON  000006         LS$MXW= 000040         L$DEA   000105         L$UNT   000003         MC$LRB  002020
LO.CSR  000006         LS$NMT= 020000         L$DLT   000110         L$XCH   000232         MC$LRP  002114
LO.CTL  000014         LS$NOR= 100000         L$DTE   000240         L$XMT   000120         MC$LRR  002006
LO.LIN  000004         LS$NTI= 000200         L$DTEL  000236         L$$ASG= 000000         MC$LSI  002032
LO.PAR  000010         LS$NTL= 000001         L$DTEP  000250         L$$DRV= 000000         MC$LST  002033
LO.PRI  000010         LS$NUM= 000100         L$FLG   000064         L$$P11= 000001         MC$LUP  006072
```

```
                                                                 J 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-3
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14     PAGE 8
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL  VALUE             REFERENCES

SF.PAC  = 000020          #5-73
SF.REA  = 000010          #5-73
SF.SER  = 000001          #5-73        24-1137    24-1140
SF.SVC  = 000002          #5-73
SF.UNL  = 000040          #5-73
SLTMA   = ******  GX       13-437
STAOFF    002100 R         13-496     #16-662
STAON     001534 R         13-490     #14-521
STASER    001706 R         13-494     #15-592
S$$BAS  = ******            7-107      21-1008
S.COST    000001          #5-73       *20-882
S.FLG     000000          #5-73        13-372     13-449     *19-813    20-883    21-983    *24-1137   *24-1140
S.LEN     000004          #5-73        19-822
S.NMST    000002          #5-73        13-377     18-732     *21-1002
S.OWNR    000003          #5-73
TH1TBL    000252 R        #8-139       22-1045
TH2TBL    000264 R        #8-140       22-1048
TH3TBL    000276 R        #8-141       22-1051
TWKLIN    002116 RG        13-478     #17-692
T$FLAG    000044          #5-68
T$LIF     000013          #5-68
T$LIFL    000013          #5-68
T$LIFO    000013          #5-68
T$LIFS    000013          #5-68
T$LIN     000000          #5-68
T$LIPS    000006          #5-68
T$LLD     000012          #5-68
T$LLDC    000045          #5-68
T$LLDL    000012          #5-68
T$LLDO    000012          #5-68
T$LLDS    000012          #5-68
T$LLEN    000046          #5-68
T$LOPR    000002          #5-68
T$LTCL    000024          #5-68
T$LTIM    000026          #5-68
T$LTPR    000014          #5-68
T$LTPS    000020          #5-68
T$NAPL    000004          #5-68
T$NFE     000000          #5-68
T$NLEN    000010          #5-68
T$NNUL    000002          #5-68
T$NOPL    000006          #5-68
T$NRNI    000042          #5-68
T$NRPL    000005          #5-68
T$NRUL    000007          #5-68
T$NVR     000001          #5-68
T$RPRI    000040          #5-68
T$SVC     000034          #5-68
T$T5      000030          #5-68       *12-303
T$T6      000032          #5-68
T.BTTC    000074            8-130
T.DELT    000056            8-132
```

```
                                            .SBTTL  SETPVC - Set PVC parameters
                                    ;+
                                    ; *** SETPVC - Set PVC parameters
                                    ;
                                    ; INPUTS
                                    ;       R4 - Address of context area
                                    ;
                                    ; OUTPUTS
                                    ;       R0 - Error code
                                    ;-

000326                              SETPVC:
000326  032764  157724  000076              BIT     #^C<LS$PVC>,L$FLX(R4)   ; Check for valid PVC grouping
000334  001013                              BNE     90$                     ; Branch if bad parameters
000336  032764  157777  000076              BIT     #^C<LS$LCT>,L$FLX(R4)   ; Anything other than COUNTER TIMER?
000344  001007                              BNE     90$                     ; If so, can't cope just yet

000346  032764  020000  000076              BIT     #LS$LCT,L$FLX(R4)       ; Check for counter timer set
000354  001402                              BEQ     10$                     ; Branch if not required
000356                                      CALL    SETLCT                  ; Set counter timer
000362                              10$:    RETURN

000364  012700  177745              90$:    MOV     #ME.GRO,R0              ; Indicate grouping error
000370  012764  177777  000072              MOV     #-1,L$PAR(R4)           ; Indicate no information
000376                                      RETURN
```

```
                                                                     I 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 1

SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE            REFERENCES

CICHPX    000200 RG      #8-145
CKCSTA    000536 R         10-220     #11-260
CKDSVC    001012 R          8-152     #14-360
C$PORT    000014           15-394
E$NBR     000014         #5-67
E$NBS     000020         #5-67
E$NCR     000034         #5-67
E$NCS     000036         #5-67
E$NIC     000044         #5-67
E$NLEN    000050         #5-67
E$NLLA    000012         #5-67
E$NLNK    000000         #5-67
E$NML     000040         #5-67
E$NMR     000024         #5-67
E$NMS     000030         #5-67
E$NNOD    000002         #5-67
E$NRT     000042         #5-67
E$NRTP    000005         #5-67
E$NSEG    000010         #5-67
E$NTIM    000046         #5-67
E$NUSE    000004         #5-67
E$STRT    000006         #5-67
FNDCTR  = ******    GX     10-223      14-363
H$PTB     000020           15-397
KISAR6  = ******    GX     10-218     *10-227     *10-238      14-362     *14-370      15-391     *15-403
LC$NTL  = 000200         #6-104
LC$OWN  = 000400         #6-104
LF$MLT  = 040000         #6-104
LF$REA  = 000001         #6-104
LF$SEG  = 100000         #6-104
LF$SKP  = 000004         #6-104
LF$VR2  = 000010         #6-104
LF$ZER  = 000002         #6-104
LF.ACT  = 100000         #5-72         11-270
LF.BRO  = 000400         #5-72
LF.BWT  = 000007         #5-72
LF.ENA  = 002000         #5-72
LF.LPB  = 001000         #5-72
LF.MDC  = 000100         #5-72
LF.MFL  = 004000         #5-72
LF.MTP  = 000020         #5-72
LF.PAC  = 000200         #5-72
LF.RDY  = 040000         #5-72         11-262
LF.REA  = 010000         #5-72
LF.SER  = 000040         #5-72
LF.TIM  = 000010         #5-72
LF.UNL  = 020000         #5-72
LF.X2P  = 000000         #5-72
LN.CLO  = 000000         #5-72
LN.DUM  = 000005         #5-72
LN.LOA  = 000004         #5-72
LN.LOO  = 000003         #5-72
```

```
                                                                     J 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 2
```

```
                                        .SBTTL  Local data

                                ;
                                ; LOCAL DATA
                                ;

000000                          NODE:   .BLKW   1               ; ADJACENT NODE ADDRESS
000002                          OWNER:  .BLKW   1               ; RAD50 OWNER PROCESS NAME
000004                          SVCMAP: .BLKW   1               ; APR BIAS FOR SVC DESCRIPTOR
000006                          SVCADD: .BLKW   1               ; VIRTUAL ADDRESS OF SVC DESCRIPTOR
000010                          PVCPRT: .BLKW   1               ; PVC PORT NUMBER
000012                          HTIM:   .BLKW   1               ; HELLO TIMER
000014                          LTIM:   .BLKW   1               ; LISTEN TIMER
000016                          BLKING: .BLKW   1               ; Blocking status
000020                          FLAG:   .BLKW   1               ; LOCAL FLAGS WORD
        000001                          FL.SVC = 1              ; DLM CIRCUIT IS AN SVC
        000002                          FL.PVC = 2              ; DLM CIRCUIT IS A PVC
        000004                          FL.INC = 4              ; DLM CIRCUIT IS INCOMING SVC
                                        .EVEN
```

```
                                          .SBTTL  SETNOD - Put node id
                                  ;+
                                  ; **-SETNOD - Put node id
                                  ;
                                  ; Inputs:       R2    = node address
                                  ;               R3    = address of next byte in message
                                  ;               (R5)  = parameter id (JSR R5)
                                  ;
                                  ; Output:       R3 is updated.  The node id, adjusted for
                                  ;               the version of NICE spoken by the connected NCP,
                                  ;               is put in the response.
                                  ;
                                  ;-

002266                            SETNOD: SAVRG   <R0,R1>                   ; save registers
002272  010201                            MOV     R2,R1                     ; copy node address
                                  ;
                                  ; Phase 4 command node: return 16 bit node address
                                  ;
002274  016700   000000G                  MOV     CURCTX,R0                 ; push current context address
002300  032760   000000C  000000G         BIT     #CF$VR2!CF$VR3,C$FLAG(R0) ; connected to phase 3 NCP ?
002306  001415                            BEQ     10$                       ; if EQ, no
                                  ;
                                  ; Phase 3 command node: mask area if target node in executor area,
                                  ;                       otherwise return 16 bit node address.
                                  ;
002310  010146                            MOV     R1,-(SP)                  ; copy address
002312  042716   001777                   BIC     #1777,(SP)                ; isolate area
002316  017700   000000G                  MOV     @DECPT,R0                 ; point to DECnet home block    ; RJK10
002322  016000   000014                   MOV     D$LNUM(R0),R0             ; get exec address
002326  042700   001777                   BIC     #1777,R0                  ; area only
002332  020026                            CMP     R0,(SP)+                  ; same area as executor ?
002334  001002                            BNE     10$                       ; br if no

002336  042701   176000                   BIC     #^C1777,R1                ; else return as area zero

002342  112523                    10$:    MOVB    (R5)+,(R3)+               ; low byte of parameter number
002344  112523                            MOVB    (R5)+,(R3)+               ; hi byte of parameter number
002346  112723   000301                   MOVB    #MT$COD!MT$MUL!1,(R3)+    ; coded multiple field (default of 1)
002352  010400                            MOV     R4,R0                     ; point to scratch buffer
002354  062700   000072                   ADD     #L$SCR,R0                 ; ...
002360  005010                            CLR     (R0)                      ; initialize buffer to unique value (0)
002362                                    CALL    MAPADD                    ; map the address into a name
002366  103403                            BCS     20$                       ; if CS, no name found
002370  112763   000302   177777          MOVB    #MT$COD!MT$MUL!2,-1(R3)   ; else, 2 fields to return
002376  112723   000002           20$:    MOVB    #2,(R3)+                  ; set address data type (2 bytes)
002402  110123                            MOVB    R1,(R3)+                  ; set low byte of address
002404  000301                            SWAB    R1                        ; ...
002406  110123                            MOVB    R1,(R3)+                  ; set high byte
002410  005710                            TST     (R0)                      ; was a name found ?
002412  001404                            BEQ     30$                       ; if EQ, no - return
002414  112723   000100                   MOVB    #MT$ASC,(R3)+             ; else, yes - set name data type
002420                                    CALL    SETNAM                    ; set name into buffer

002424                            30$:    RESRG   <R1,R0>                   ; restore registers
002430  000205                            RTS     R5
```

```
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF    04.00

SYMBOL  VALUE              REFERENCES

AT$UP   = 000010            14-533      15-632
A$CIR     000003            15-634
A$LEN     000004            15-629
A$NID     000000            14-535      15-637
A$TMI     000002            14-538      15-640
A$TYP     000002            14-533      15-632
BCDASC  = ******   GX       19-867
BF.TRC  = 000004           #6-74
BIT0    = 000001           #7-81
BIT1    = 000002           #7-81
BIT10   = 002000           #7-81
BIT11   = 004000           #7-81
BIT12   = 010000           #7-81
BIT13   = 020000           #7-81
BIT14   = 040000           #7-81
BIT15   = 100000           #7-81
BIT2    = 000004           #7-81
BIT3    = 000010           #7-81
BIT4    = 000020           #7-81
BIT5    = 000040           #7-81
BIT6    = 000100           #7-81
BIT7    = 000200           #7-81
BIT8    = 000400           #7-81
BIT9    = 001000           #7-81
BLKING    000016 R         #8-96        9-234      *13-479     *13-482
B$10      000003           #6-81
B$11      000004           #6-81
B$12      000005           #6-81
B$5       000000           #6-81
B$6       000001           #6-81
B$7       000002           #6-81
B$7S10    000006           #6-81
CF$VR2  = ******   GX       16-667
CF$VR3  = ******   GX       16-667
CIRECH    000022 RG        #9-122
CURCTX  = ******   GX       16-666
C$FLAG  = ******   GX       16-667
DECPT   = ******   GX       9-174       14-546      14-560      16-675
DF.CRC  = 000010           #6-74
DF.FOC  = 000020           #6-74
DF.RAN  = 000040           #6-74
DF.UNK  = 000100           #6-74
DMPCTB  = ******   GX       11-394
D$LNUM    000014            9-175       16-676
D$NBEA    000056            14-562
D$NBRA    000054            14-561
D$NLN     000030            14-547      14-563
E$NBR     000014           #6-63
E$NBS     000020           #6-63
E$NCR     000034           #6-63
E$NCS     000036           #6-63
E$NIC     000044           #6-63
```

```
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16     PAGE 2
```

```
                                        .SBTTL  MACRO CALLS AND LOCAL DEFINITIONS

                               ;
                               ; MACRO LIBRARY CALLS
                               ;
                                        .MCALL  SAVMAP,MAP,BIAS,RESMAP,SAVRG,RESRG
                                        .MCALL  MANDF$,SLTDF$,PDVDF$,CIRCX$

      000000                            MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
      000000                            SLTDF$                          ; DEFINE SLT OFFSETS
      000000                            PDVDF$                          ; DEFINE PDV OFFSETS

```

```
                                                                                  I 15
CIRECO      CREATED BY  MACRO  ON 29-JUN-85 AT 12:17      PAGE 2

SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL   VALUE              REFERENCES

LF.X2P   = 000000           #5-63
LN.CLO   = 000000           #5-63
LN.DUM   = 000005           #5-63
LN.LOA   = 000004           #5-63
LN.LOO   = 000003           #5-63
LN.OAU   = 000003           #5-63
LN.OFF   = 000001           #5-63
LN.ON    = 000000           #5-63
LN.OOP   = 000004           #5-63
LN.OPE   = 000001           #5-63
LN.REF   = 000002           #5-63
LN.SER   = 000002           #5-63
LN.STA   = 000017           #5-63
LN.SUB   = 000360           #5-63
LN.TRI   = 000006           #5-63
LP$MPT   = 000010           #7-69
LP$MUX   = 000004           #7-69
LP$TRB   = 000002           #7-69
LP$UNT   = 000001           #7-69
LP$WCN   = 000040           #7-69         7-69
LP$WDV   = 000020           #7-69         7-69
LP$WLD   = 000360           #7-69
LP$WTR   = 000200           #7-69         7-69
LP$WUN   = 000100           #7-69         7-69
L$ADJ      000110           #6-69
L$BUF      000060           #6-69         10-167
L$CHN      000054           #6-69
L$COU      000114           #6-69
L$CTB      000053           #6-69
L$CTL      000002           #6-69
L$DDM      00000C           #6-69
L$FLG      000064           #6-69
L$FLX      000070           #6-69
L$LEN      000124           #6-69
L$LTM      000070           #6-69
L$MSG      000056           #6-69
L$MTYP     000067           #6-69
L$NAM      000000           #6-69
L$NLEN   = 000020           #6-69         6-69          6-69
L$NOD      000112           #6-69
L$NXT      000044           #6-69
L$OPT      000062           #6-69
L$PAR      000120           #6-69
L$PDV      000055           #6-69
L$PFG      000041           #6-69
L$PLB      000116           #6-69
L$PRO      000067           #6-69
L$PVC      000004           #6-69
L$SCN      000020           #6-69
L$SCR      000072           #6-69
L$SLEN     000066           #6-69
L$SLT      000042           #6-69         10-138
```

```
Symbol table

S2.CTO= 040000            S5.BCC= 020000            US.MDM= 000020            U.FNUM= 000040            U.VCB = 000034
S2.CTS= 100000            S5.DAO= 040000            US.MNT= 000100            U.GRP = 000102            U.VSER= 000120
S2.ELF= 001000            S5.HPC= 000014            US.OFL= 000001            U.KCSR= 000032            U.2MED= 000070
S2.FLF= 000400            S5.HPO= 000020            US.PUB= 000004            U.KCS6= 000034            U2.AT.= 000020
S2.IRQ= 000200            S5.ITI= 000100            US.PWF= 000010            U.LUIC  177774            U2.CRT= 002000
S2.OBF= 004000            S5.OXF= 000040            US.RED= 000002            U.MEDI= 000040            U2.DH1= 100000
S2.ORQ= 000100            S5.RPO= 002000            US.SHR= 000001            U.MLUN= 000050            U2.DJ1= 040000
S2.PCU= 010000            S5.SW1= 000001            US.SPU= 000002            U.MUP   177772            U2.DZ1= 000100
S2.RCU= 000001            S5.TMM= 000002            US.UMD= 000010            U.OWN   177776            U2.ESC= 001000
S2.SRQ= 000040            S5.VER= 010000            US.VV = 000001            U.RBNS= 000112            U2.HFF= 010000
S2.WAL= 000010            S5.YOF= 000004            US.WCK= 000010            U.RCTC= 000113            U2.HLD= 000040
S2.WRA= 000006            S5.XON= 000010            UU.ABO= 000400            U.RCTS= 000110            U2.LOG= 000400
S2.WRB= 000002            S6.EIO= 000400            UU.ATN= 000100            U.RED   000002            U2.LWC= 000001
S3.ACR= 000001            S6.RDI= 100000            UU.AVN= 000004            U.RED2  000034            U2.L3S= 000004
S3.CTC= 000004            S6.RLU= 001000            UU.GUS= 000010            U.SCB   000020            U2.L8S= 010000
S3.FDX= 000200            T$$KMG= 000000            UU.IOS= 002000            U.SHST= 000076            U2.NEC= 004000
S3.ICE= 001000            T$$MIN= 000000            UU.ONL= 000020            U.SHUN= 000074            U2.PRV= 000010
S3.MHE= 000400            UC.ALG= 000200            UU.RCT= 000002            U.SPC = 000036            U2.RMT= 020000
S3.NEC= 000020            UC.ATT= 000010            UU.RDY= 000200            U.STS   000005            U2.R04= 100000
S3.PPT= 020000            UC.KIL= 000004            UU.SER= 000001            U.ST2   000007            U2.SCS= 000004
S3.PTH= 004000            UC.LGH= 000003            UU.SIO= 001000            U.SUB = 000036            U2.SLV= 000200
S3.RAL= 000010            UC.NPR= 000100            UU.SPC= 000040            U.TCHP  000042            U2.VT5= 000002
S3.RES= 010000            UC.PWF= 000020            U.ACB   000062            U.TCVP  000043            U2.7CH= 010000
S3.RUB= 040000            UC.QUE= 000040            U.ACP = 000032            U.TFLK  000040            U3.DPA= 100000
S3.TAB= 000002            UD.UNS= 000000            U.ADMA  000066            U.TFRQ  000036            U3.PAR= 040000
S3.TME= 002000            UD.160= 000004            U.AFLG  000064            U.TIXL  000060            U3.UPC= 020000
S3.TSY= 000040            UD.200= 000001            U.ATT   000022            U.TLPP  000053            U4.CR = 000100
S3.8BC= 000100            UD.556= 000002            U.BPKT= 000044            U.TMTI  000047            V$$CTR= 001000
S4.ABD= 100000            UD.625= 000005            U.BUF   000024            U.TRCK= 000100            W.BATT  000006
S4.ANI= 000400            UD.8K = 000006            U.CBF = 000032            U.TSTA  000026            W.BFPD  000014
S4.AVO= 001000            UD.800= 000003            U.CLI   177772            U.TST5  000054            W.BHVR  000004
S4.BLK= 002000            UM.CLI= 000036            U.CNT   000030            U.TST6  000056            W BLGH  000020
S4.DEC= 004000            UM.DSB= 000200            U.COTQ  000030            U.TTAB  000050            W.BLPD  000016
S4.DLO= 000100            UM.NBR= 000400            U.CTCB  000026            U.TTYP  000046            W.BLVR  000002
S4.EDT= 010000            UM.OVR= 000001            U.CTL   000004            U.TUX   000024            W.BNPD  000015
S4.HFF= 000020            US.ABO= 000001            U.CTYP  000052            U.UHVR= 000107            W.BOFF  000012
S4.HFL= 000007            US.BSP= 000002            U.CW1   000010            U.UIC   000044            W.BPCB  000000
S4.HHT= 000040            US.BSY= 000200            U.CW2   000012            U.UNFL= 000052            W.BSIZ  00001C
S4.HSY= 000200            US.FOR= 000040            U.CW3   000014            U.UNIT  000006            X$$DBT= 000000
S4.RGS= 020000            US.FRK= 000002            U.CW4   000016            U.UNSZ= 000114            $MBLUN= ****** GX
S4.SFC= 040000            US.KPF= 000001            U.CYL = 000104            U.UNTI= 000060            $SAVRG= ****** GX
S4.VFL= 000010            US.LAB= 000004            U.DCB   000000            U.USVR= 000106            $$$SYS= 004374
S5.ABP= 100000            US.MDE= 000002            U.FCDE= 000042            U.UTIL= 000036            $$$SYX= 000000


. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        000224    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 227
Work  file writes: 154
Size of work file: 30160 Words  ( 118 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS
```

```
                                         J  1
CICHIN   CREATED BY  MACRO  ON 29-JUN-85 AT 12:11    PAGE 4
SYMBOL CROSS REFERENCE                               CREF   04.00

SYMBOL   VALUE          REFERENCES

LS$LCT  = 020000        #6-72     6-72
LS$LMB  = 000002        #6-72
LS$LOO  = 040000        #6-72
LS$MDT  = 020000        #6-72
LS$MRT  = 000020        #6-72
LS$MWN  = 000040        #6-72
LS$MXB  = 001000        #6-72
LS$MXR  = 000020        #6-72     6-72      6-72
LS$MXW  = 000040        #6-72     6-72      6-72      6-72
LS$NMT  = 020000        #6-72
LS$NOR  = 100000        #6-72
LS$NTI  = 000200        #6-72
LS$NTL  = 000001        #6-72
LS$NUM  = 000100        #6-72     6-72
LS$OWN  = 000010        #6-72
LS$PLT  = 004000        #6-72
LS$PRO  = 010000        #6-72
LS$PVC  = 020053        #6-72
LS$RET  = 000200        #6-72     6-72      6-72
LS$SER  = 100000        #6-72
LS$STA  = 000002        #6-72
LS$SVC  = 000362        #6-72
LS$TAD  = 000020        #6-72
LS$TH1  = 000100        #6-72
LS$TH2  = 000200        #6-72
LS$TH3  = 000400        #6-72
LS$XMT  = 040000        #6-72
LS.AST  = 100000        #5-64
LS.CEX  = 000001        #5-64
LS.CIR  = 000010        #5-64
LS.CXO  = 010000        #5-64
LS.CX5  = 000400        #5-64
LS.DLM  = 004000        #5-64
LS.ECH  = 001000        #5-64
LS.FDX  = 004000        #5-64
LS.HDX  = 010000        #5-64
LS.LIN  = 000003        #5-64
LS.LMC  = 000007        #5-64
LS.NTI  = 004000        #5-64
LS.ON   = 000011        #5-64
LS.OPT    000400        #5-64
LS.PRO  = 000002        #5-64
LS.PWF  = 040000        #5-64
LS.TOP  = 002000        #5-64
LS.UNF  = 020000        #5-64
LS.X25  = 000012        #5-64
LS.11D  = 004000        #5-64
LX.CEX  = 000004        #5-64
LX.LIN  = 000006        #5-64
LX.PRO  = 000005        #5-64
L$ACT     000104        #6-72    *17-815
L$BBT     000114        #6-72    *29-1089
```

```
                                         K  1
CICHIN   CREATED BY  MACRO  ON 29-JUN-85 AT 12:11    PAGE 5
```

```
                                         .SBTTL  Context area definition

       000000                            CICCX$  LIST
                                     ;
                                     ; CONTEXT AREA OFFSET DEFINITIONS
                                     ;
       000004                            .ASECT
                   000000                . = 0
                   000020                L$NLEN  = 16.          ; MAXIMUM NAME LENGTH
                                     ;
                                     ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                     ; as in the CIRCX$ and LOCCX$ macros.
                                     ;
       000000                        L$NAM:
       000000                        L$DDM:  .BLKW   1              ; DEVICE NAME
       000002                        L$CTL:  .BLKB   1              ; CONTROLLER NUMBER
       000003                        L$UNT:  .BLKB   1              ; UNIT NUMBER
       000004                        L$PVC:  .BLKW   1              ; 3RD WORD OF PVC NAME
                   000020                    . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
       000020                        L$SCN:  .BLKB   L$NLEN         ; CURRENT NAME IN WILDCARD SCAN
       000040                        L$TRB:  .BLKB   1              ; TRIBUTARY NUMBER
       000041                        L$PFG:  .BLKB   1              ; PARSE FLAGS
       000042                        L$SLT:  .BLKW   1              ; Current SLT/PVC address
       000044                        L$NXT:  .BLKW   1              ; Pointer to next SLT/PVC address
       000046                        L$SNM:  .BLKW   1              ; REMAINING NUMBER OF SYSTEM LINES
       000050                        L$TPT:  .BLKW   1              ; Current tributary pointer (-1 for PSI)
       000052                        L$TNM:  .BLKB   1              ; REMAINING NUMBER OF TRIBUTARIES
       000053                        L$CTB:  .BLKB   1              ; CURRENT TRIBUTARY NUMBER
       000054                        L$CHN:  .BLKB   1              ; Channel number / X.25 port number and
       000055                        L$PDV:  .BLKB   1              ; ... PDV assigned
       000056                        L$MSG:  .BLKW   1              ; ERROR MESSAGE STRING POINTER
       000060                        L$BUF:  .BLKW   1              ; SAVED BUFFER POINTER
       000062                        L$OPT:  .BLKB   1              ; SAVED OPTIONS BYTE
       000063                        L$TYP:  .BLKB   1              ; CIRCUIT-ID FORMAT TYPE
       000064                        L$FLG:  .BLKW   1              ; FLAG WORD (FOR COMMAND)
       000066                        L$FL1:  .BLKW   1              ; FLAG WORD (FOR COMMAND)
       000070                        L$FL2:  .BLKW   1              ; FLAG WORD (FOR CURRENT CIRCUIT)
       000072                        L$PAR:  .BLKW   1              ; CURRENT PARAMETER TYPE
       000074                        L$SLEN: .BLKB   1              ; Significant length of circuit name
       000075                        L$PRO:                         ; Line protocol
       000075                        L$MTYP: .BLKB   1              ; Network management circuit type
       000076                        L$FLX:  .BLKW   1              ; Flags word for X.25 circuit commands
       000100                        L$STA:  .BLKB   1              ; CIRCUIT STATE
       000101                        L$COS:  .BLKB   1              ; CIRCUIT COST
       000102                        L$OWN:  .BLKB   1              ; CIRCUIT OWNER (PDV INDEX)
       000103                        L$TAD:  .BLKB   1              ; TRIBUTARY ADDRESS
       000104                        L$ACT:  .BLKB   1              ; MULTIPOINT ACTIVE RATIO
       000105                        L$DEA:  .BLKB   1              ; MULTIPOINT DEAD RATIO
       000106                        L$DDT:  .BLKW   1              ; DMP DEAD TIMER
       000110                        L$DLT:  .BLKW   1              ; DMP DELAY TIMER
       000112                        L$PLT:  .BLKW   1              ; DMP POLL TIMER
       000114                        L$BBT:  .BLKW   1              ; DMP BABBLE TIMER
       000116                        L$NMT:  .BLKW   1              ; DMP NORMAL TIMER
       000120                        L$XMT:  .BLKW   1              ; DMP TRANSMIT TIMER
       000122                        L$BSA:  .BLKB   1              ; DMP ACTIVE BASE
       000123                        L$BSD:  .BLKB   1              ; DMP DYING BASE
       000124                        L$BSI:  .BLKB   1              ; DMP INACTIVE BASE
```

```
                                                                  J  3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-2
Symbol table

A$$CHK= 000000            LF$MLT= 040000            LR.EFN  000004            LS.CXO= 010000            L$MXB   000133
A$$CPS= 000000            LF$REA= 000001            LR.LIN  000002            LS.CX5= 000400            L$MXR   000253
A$$PRI= 000000            LF$SEG= 100000            LR.PRM  000030            LS.DLM= 004000            L$MXW   000252
A$$TRP= 000000            LF$SKP= 000004            LR.PRO  000002            LS.ECH= 001000            L$NAM   000000
CICHOP  000134RG          LF$VR2= 000010            LR.STS  000000            LS.FDX= 004000            L$NLEN= 000020
CICHPX= ****** GX         LF$ZER= 000002            LR.TCB  000002            LS.HDX= 010000            L$NMT   000116
CLENTL= ****** GX         LF.ACT= 100000            LR.UNT  000005            LS.LIN= 000003            L$NTI   000256
CLNUP   000552R           LF.BRO= 000400            LS$ACT= 000040            LS.LMC= 000007            L$NTL   000134
C$$CKP= 000000            LF.BWT= 000007            LS$BBT= 010000            LS.NTI= 004000            L$NUM   000240
C$$ORE= 000400            LF.ENA= 002000            LS$BLK= 001000            LS.ON = 000011            L$NUML  000236
C$$RSH= 177564            LF.LPB= 001000            LS$BSA= 000001            LS.OPT= 000400            L$NXT   000044
DISTBL  000000R           LF.MDC= 000100            LS$BSD= 000002            LS.PRO= 000002            L$OPT   000062
DISTB1  000056R           LF.MFL= 004000            LS$BSI= 000004            LS.PWF= 040000            L$OWN   000102
DSPTCH  000440R           LF.MTP= 000020            LS$CHN= 000001            LS.TOP= 002000            L$PAR   000072
D$$BUG= 177514            LF.PAC= 000200            LS$CMB= 000002            LS.UNF= 020000            L$PDV   000055
D$$ISK= 000000            LF.RDY= 040000            LS$COS= 000004            LS.X25= 000012            L$PFG   000041
D$$L11= 000001            LF.REA= 010000            LS$CUS= 000004            LS.11D= 004000            L$PLT   000112
D$$YNC= 000000            LF.SER= 000040            LS$DDT= 001000            LX.CEX= 000004            L$PRO   000075
D$$YNM= 000000            LF.TIM= 000010            LS$DEA= 000100            LX.LIN= 000006            L$PVC   000004
E$NBR   000014            LF.UNL= 020000            LS$DLM= 001004            LX.PRO= 000005            L$RET   000256
E$NBS   000020            LF.X2P= 000000            LS$DLT= 002000            L$ACT   000104            L$SCN   000020
E$NCR   000034            LN.CLO= 000000            LS$DTE= 000010            L$BBT   000114            L$SCR   000166
E$NCS   000036            LN.DUM= 000005            LS$HBT= 000400            L$BLK   000262            L$SER   000230
E$NIC   000044            LN.LOA= 000004            LS$HTM= 010000            L$BSA   000122            L$SLEN  000074
E$NLEN  000050            LN.LOD= 000003            LS$INA= 000010            L$BSD   000123            L$SLT   000042
E$NLLA  000012            LN.OAU= 000003            LS$IND= 000020            L$BSI   000124            L$SNM   000046
E$NLNK  000000            LN.OFF= 000001            LS$INI= 000040            L$BUF   000060            L$STA   000100
E$NML   000040            LN.ON = 000000            LS$LCT= 020000            L$CHN   000054            L$TAD   000103
E$NMR   000024            LN.OOP= 000004            LS$LMB= 000002            L$CMB   000234            L$TH1   000130
E$NMS   000030            LN.OPE= 000001            LS$LOO= 040000            L$COS   000101            L$TH2   000131
E$NNOD  000002            LN.REF= 000002            LS$MDT= 020000            L$CTB   000053            L$TH3   000132
E$NRT   000042            LN.SER= 000002            LS$MRT= 000020            L$CTL   000002            L$TNM   000052
E$NRTP  000005            LN.STA= 000017            LS$MWN= 000040            L$CUS   000260            L$TPT   000050
E$NSEG  000010            LN.SUB= 000360            LS$MXB= 001000            L$DDM   000000            L$TRB   000040
E$NTIM  000046            LN.TRI= 000006            LS$MXR= 000020            L$DDT   000106            L$TYP   000063
E$NUSE  000004            LO.CON  000006            LS$MXW= 000040            L$DEA   000105            L$UNT   000003
E$STRT  000006            LO.CSR  000006            LS$NMT= 020000            L$DLT   000110            L$XCH   000232
E$$XPR= 000000            LO.CTL  000014            LS$NOR= 100000            L$DTE   000240            L$XMT   000120
FMTLIN= ****** GX         LO.LIN  000004            LS$NTI= 000200            L$DTEL  000236            L$$ASG= 000000
FNDCTR  000622RG          LO.PAR  000010            LS$NTL= 000001            L$DTEP  000250            L$$DRV= 000000
F$$LVL= 000001            LO.PRI  000010            LS$NUM= 000100            L$FLG   000064            L$$P11= 000001
G$$TPP= 000000            LO.SCR  000014            LS$OWN= 000010            L$FLX   000076            L$$11R= 000000
G$$TSS= 000000            LO.STA  000015            LS$PLT= 004000            L$FL1   000066            L.COST  000015
G$$TTK= 000000            LO.VCT  000012            LS$PRO= 010000            L$FL2   000070            L.CTL   000012
G$$WRD= 000000            LP$MPT= 000010            LS$PVC= 020053            L$HBT   000254            L.CVA   177776
I$$RAR= 000000            LP$MUX= 000004            LS$RET= 000200            L$HTM   000226            L.DDM   000002
I$$RDN= 000000            LP$NXC= 100000            LS$SER= 100000            L$INA   000125            L.DDS   000004
KISAR6= ****** GX         LP$TRB= 000002            LS$STA= 000002            L$IND   000126            L.DLC   000003
K$$CNT= 177546            LP$UNT= 000001            LS$SVC= 000362            L$INI   000127            L.DLM   000006
K$$CSR= 177546            LP$WCN= 000040            LS$TAD= 000020            L$LCT   000224            L.DLS   000010
K$$LDC= 000000            LP$WDV= 000020            LS$TH1= 000100            L$LEN   000264            L.FLG   000000
K$$TPS= 000074            LP$WLD= 000360            LS$TH2= 000200            L$LMB   000234            L.KRBA  000016
LC$EXT  000000            LP$WTR= 000200            LS$TH3= 000400            L$LTM   000230            L.LEN = 000022
LC$INT  000001            LP$WUN= 000100            LS$XMT= 040000            L$MRT   000253            L.MPF   000022
LC$NTL= 000200            LR.AST  000026            LS.AST= 100000            L$MSG   000056            L.NMST  000020
LC$OWN= 000400            LR.CIR  000002            LS.CEX= 000001            L$MTYP  000075            L.NSTA  000014
LD$LP = 000000            LR.CTL  000004            LS.CIR= 000010            L$MWN   000252            L.OWNR  000021
```

```
                                                                  K  3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-3
```

```
                                                                  J  4
CICHOP      CREATED BY  MACRO  ON 29-JUN-85 AT 12:13     PAGE 9
MACRO CROSS REFERENCE                                    CREF    04.00

MACRO NAME       REFERENCES

BIAS             #6-60       13-418
CALL              9-180      9-184      9-186      9-198      9-204      9-211      9-254      10-282     10-288     12-363
CICCX$           #6-60       7-73
CTRDF$           #6-60       6-64
MANDF$           #6-60       6-65
MAP              #6-60       13-409     13-415
NTLDF$           #6-60       6-66
PDVDF$           #6-61       6-67
RESRG            #6-61       12-365
RETURN            9-261      10-290     11-331     12-366     13-428
SAVRG            #6-61       12-356
SLTDF$           #6-61       6-68
XPDDB$           #6-61       6-69
```

```
                                                                  K  4
**FILE**ID**CICHPD
```

```
                                                  .SBTTL  NTLLIN - PERFORM NTL SET/CLEAR LINE FUNCTIONS
                                          ;+
                                          ; **-NTLLIN-PERFORM NTL SET/CLEAR LINE FUNCTIONS
                                          ;
                                          ; THIS ROUTINE MANAGES THE SEND/RECEIVE INTERFACE TO NTL TO SET
                                          ; AND CLEAR LINES.
                                          ;
                                          ; INPUTS:
                                          ;       R4 = POINTER TO CONTEXT AREA
                                          ;
                                          ; OUTPUTS:
                                          ;       IF CC, SUCCESS, ELSE,
                                          ;       ERROR CODE IN R0
                                          ;
                                          ; REGISTERS:
                                          ;       R4, R5 ARE PRESERVED
                                          ;-

 000446                                   NTLLIN:
 000446                                           SAVRG   <R4>                            ; SAVE R4
 000450  016464  000000  000136                   MOV     L$DDM(R4),L$NTL+LR.LIN(R4)      ; SET LIN DDM NAME
 000456  116464  000002  000140                   MOVB    L$CTL(R4),L$NTL+LR.CTL(R4)      ; SET CONTROLLER NUMBER
 000464  116464  000003  000141                   MOVB    L$UNT(R4),L$NTL+LR.UNT(R4)      ; SET UNIT NUMBER
 000472  010400                                   MOV     R4,R0                           ; COPY CONTEXT BLOCK ADDRESS
 000474  010402                                   MOV     R4,R2                           ; ...
 000476  062700  000134                           ADD     #L$NTL,R0                       ; POINT TO NTL MESSAGE
 000502  062704  000166                           ADD     #L$SCR,R4                       ; POINT TO SCRATCH SEND BUFFER
 000506  010403                                   MOV     R4,R3                           ; ...
 000510  012701  000015                           MOV     #13.,R1                         ; SET 13. WORDS TO MOVE
 000514  012023                           5$:     MOV     (R0)+,(R3)+                     ; MOVE MESSAGE INTO SEND/RECEIVE BUFFER
 000516                                           SOB     R1,5$                           ; ...
 000522                                           CALL    NTLSND                          ; SEND IT TO NTL
 000526  103004                                   BCC     10$                             ; IF CC, CONTINUE

                                                  .IF NDF R$$PRO  ; PRO/DECNET

 000530  012762  000000' 000056                   MOV     #INSMSG,L$MSG(R2)               ; ELSE, SET ''INSTALLED'' ERROR

                                                  .ENDC   ; NDF R$$PRO

 000536  000406                                   BR      20$                             ; AND RETURN
 000540  005764  000004                   10$:    TST     4(R4)                           ; DID THE NTL FUNCTION SUCCEED ?
 000544  100006                                   BPL     30$                             ; IF PL, SUCCESS

                                                  .IF NDF R$$PRO  ; PRO/DECNET

 000546  012762  000035' 000056                   MOV     #NTLMSG,L$MSG(R2)               ; ELSE, NTL FUNCTION FAILED

                                                  .ENDC   ; NDF R$$PRO

 000554  012700  177747                   20$:    MOV     #ME.OPE,R0                      ; SAY OPERATION FAILED
 000560  000261                                   SEC                                     ; RETURN WITH ERROR
 000562                                   30$:    RESRG   <R4>                            ; RESTORE R4
 000564                                           RETURN

                                                  .ENDC   ; NDF R$$11S & S$$BAS & R$$RTR

```

```
                                        .SBTTL  LINON - TURN CIRCUIT ON
                                ;+
                                ; **-LINON-TURN CIRCUIT ON
                                ;
                                ; THIS ROUTINE IS CALLED TO TURN A CIRCUIT ON
                                ;
                                ; INPUTS:
                                ;       R4 = ADDRESS OF CONTEXT AREA
                                ;
                                ; OUTPUTS:
                                ;       IF CC, CIRCUIT TURNED ON,
                                ;       ELSE, ERROR RETURN IN R0
                                ;-

002174                          LINON:
                                ;
                                ; CHECK THAT CIRCUIT IS IN CORRECT STATE
                                ;
002174  012700  177765                  MOV     #ME.CST,R0              ; ASSUME CIRCUIT IS IN WRONG STATE
002200  012764  000003  000072          MOV     #ME$CIR,L$PAR(R4)       ; ...
002206  016401  000042                  MOV     L$SLT(R4),R1            ; GET SLT ADDRESS
002212  105761  000014                  TSTB    L.NSTA(R1)              ; MULTI-POINT/ETHEPNET CIRCUIT?
002216  001405                          BEQ     5$                      ; BR IF NO
002220  016401  000050                  MOV     L$TPT(R4),R1            ; POINT TO CURRENT TRIB TABLE
002224  062701  000002                  ADD     #S.NMST,R1              ; POINT TO MANAGEMENT STATE BYTE
002230  000402                          BR      7$                      ; CONTINUE
002232  062701  000020          5$:     ADD     #L.NMST,R1              ; POINT TO MANAGEMENT STATE BYTE
002236  122711  000001          7$:     CMPB    #LN.OFF,(R1)            ; IS THE CIRCUIT OFF?
002242  001062                          BNE     60$                     ; IF NE, NO - ERROR
                                ;
                                ; SET CIRCUIT ON
                                ;
002244                          10$:    SWSTK$  30$                     ;; ENTER SYSTEM STATE
002250  010405                          MOV     R4,R5                   ;; SAVE ADDRESS OF CONTEXT AREA
002252                                  CALL    @CCBGT                  ;; TRY TO ALLOCATE A CCB        ; RJK08
002256  103420                          BCS     20$                     ;; IF CS, ALLOCATION FAILURE
002260  116564  000055  000007          MOVB    L$PDV(R5),C.STA(R4)     ;; SET UP PDV AND
002266  116564  000054  000006          MOVB    L$CHN(R5),C.LIN(R4)     ;;   CHANNEL # TO TURN ON
002274  005064  000004                  CLR     C.NSP(R4)               ;; SET RETURN TO AUX
002300  012764  000004  000012          MOV     #4,C.STS(R4)            ;; TURN ON THE CIRCUIT
002306  012764  001006  000010          MOV     #FC.CTL!FS.STR,C.FNC(R4);; SET UP FUNCTION CODE
002314                                  CALLR   @LLCRS                  ;; QUEUE CCB TO NETACP         ; RJK08

002320                          20$:    RETC    R0                      ;; SET USER C-BIT
002332                                  RETURN                          ;; LEAVE SYSTEM STATE

002334  103004                  30$:    BCC     40$                     ; IF CC, REQUEST QUEUED
002336                                  WSIG$S                          ; ALLOCATION FAILURE ... WAIT A BIT
002344  000737                          BR      10$                     ;   THEN TRY AGAIN

002346  016401  000042          40$:    MOV     L$SLT(R4),R1            ; GET SLT ADDRESS FOR CIRCUIT
002352  105761  000014                  TSTB    L.NSTA(R1)              ; IS THIS MULTI-DROP/ETHERNET ?
002356  001410                          BEQ     50$                     ; IS EQ, NO
002360  116400  000040                  MOVB    L$TRB(R4),R0            ; GET TRIBUTARY NUMBER
002364  006300                          ASL     R0                      ; MULTIPLY IT BY 4
002366  006300                          ASL     R0                      ; ...
002370  060100                          ADD     R1,R0                   ; POINT AT THE TRIB
```

```
                                                                  J 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-3
Symbol table

MC$MAP  010000          MC$XLJ  000323          MD.NON  177777          MF$ACT  177776          MO$PRO  000002
MC$MBL  001764          MC$XLR  002330          MD.PER  000000          MF$ADD  000000          MO$PRS  000001
MC$MBX  004420          MC$XMA  000310          MD.RES  000004          MF$ADJ  177774          MO$REA  000200
MC$MBY  001752          MC$XMC  002305          MD.ROB  000005          MF$ALL  177775          MO$SET  000000
MC$MPX  004406          MC$XMS  002304          MD.RSH  000013          MF$BYE  000300          MO$STA  000020
MC$NAP  001604          MC$XNR  002332          MD.SDU  000005          MF$CHA  000023          MO$SUM  000000
MC$NBR  001130          MC$XRC  002316          MD.SLO  000003          MF$DUM  000020          MO$TSK  000000
MC$NBS  001131          MC$XRR  002331          MD.SNA  000000          MF$EVT  000001          MO$VOL  000000
MC$NCR  001154          MC$XRS  002342          MD.TLO  000004          MF$FUP  000264          MO$ZER  000000
MC$NCS  001155          MC$020  000002          MD.UNA  000002          MF$KNO  177777          MP$ACB  000012
MC$NML  001274          MD$CI   000007          MD.UNR  000003          MF$LOA  000017          MP$ACC  000514
MC$NMR  001142          MD$CNA  000003          MD.UOB  000007          MF$LOO  177775          MP$ACT  004526
MC$NMS  001143          MD$COU  000001          MD.VOL  000006          MF$REA  000024          MP$ADD  000766
MC$NNO  001606          MD$DA   000010          ME$ALI  000010          MF$SIG  177773          MP$ADJ  001440
MC$NNU  001605          MD$DL   000004          ME$AL2  000003          MF$SPF  000302          MP$ADP  006325
MC$NOP  001607          MD$DLV  000020          ME$ARE  000005          MF$SYS  000026          MP$ADS  004406
MC$NOR  000000          MD$DMC  000014          ME$CIR  000003          MF$TES  000022          MP$ALB  000036
MC$NPF  001616          MD$DMF  000046          ME$EXA  000000          MF$TRA  000301          MP$AMC  001640
MC$NPR  001630          MD$DMP  000022          ME$EXE  000200          MF$TRI  000021          MP$AMH  001641
MC$NRE  001200          MD$DMR  000050          ME$LIN  000001          MF$ZER  000025          MP$ANB  004420
MC$NRT  001166          MD$DMV  000042          ME$LOG  000002          ML$ALL  100000          MP$ASB  000024
MC$NVR  001642          MD$DN   000016          ME$MOD  000004          ML$CLS  000000          MP$ASC  004432
MC$P08  000000          MD$DP   000000          ME$NOD  000000          ML$CON  000001          MP$AUS  000512
MC$P11  000001          MD$DPV  000044          ME$NON  177777          ML$EXT  000001          MP$BBT  002165
MC$RFL  002046          MD$DQ   000006          ME$OBJ  000007          ML$FIL  000002          MP$BDF  002164
MC$SBU  002051          MD$DTE  000024          ME$OB2  000004          ML$FIR  000000          MP$BFQ  002121
MC$SFL  002044          MD$DU   000002          ME$PRO  000005          ML$INT  000000          MP$BLK  001616
MC$SYC  000012          MD$DUP  000012          ME$SYS  000006          ML$KNO  140000          MP$BLO  001452
MC$SYL  000036          MD$DV   000026          ME.BLO  177744          ML$MON  000003          MP$BMX  002176
MC$SYR  000050          MD$DZ   000030          ME.CON  177753          ML$SYS  000002          MP$BNP  006330
MC$SYS  000024          MD$FUL  000000          ME.CST  177765          ML$TOP  000001          MP$BRT  001620
MC$SYZ  000000          MD$HAL  000001          ME.DIS  177755          MN$UNL  000377          MP$BSA  002176
MC$TBR  001140          MD$HEL  000002          ME.DON  177600          MO$ACC  000200          MP$BSD  002203
MC$TBS  001141          MD$KCP  000013          ME.FCO  177762          MO$ADD  000002          MP$BSI  002200
MC$TDR  002053          MD$KDP  000034          ME.FIO  177756          MO$ALA  000004          MP$BSP  006331
MC$TIM  000000          MD$KDZ  000036          ME.FOP  177763          MO$ALI  000000          MP$BUF  000170
MC$TYP  007777          MD$KL   000040          ME.FOR  177776          MO$CHA  000040          MP$BUP  006324
MC$UBU  002052          MD$KMX  000054          ME.FUN  177777          MO$CIR  000003          MP$BUS  001643
MC$UFD  002047          MD$KMY  000052          ME.GRO  177745          MO$CLE  000100          MP$CAC  001750
MC$UMR  001132          MD$LEN  000050          ME.HAR  177750          MO$COU  000060          MP$CAS  001762
MC$UMS  001133          MD$PCL  000011          ME.IID  177767          MO$DAC  000000          MP$CAT  002210
MC$VAX  000003          MD$QNA  000005          ME.LCO  177766          MO$DEF  000000          MP$CCS  004406
MC$WID  060000          MD$UNA  000001          ME.LPR  177757          MO$DPR  000000          MP$CHN  002141
MC$WIH  040000          MD$WIT  000002          ME.MPR  177773          MO$ENT  000017          MP$CIR  000144
MC$WIL  020000          MD.ABO  000017          ME.MVE  177771          MO$EVE  000100          MP$CLK  002131
MC$W08  020000          MD.ACC  000010          ME.OPE  177747          MO$INF  000160          MP$CLN  002126
MC$W16  040000          MD.AOB  000016          ME.PLO  177751          MO$INS  000002          MP$CLT  002211
MC$W32  060000          MD.BOB  000011          ME.PMI  177743          MO$LIN  000001          MP$CMB  002142
MC$XBR  001750          MD.DIA  000007          ME.PNA  177752          MO$LOG  000001          MP$CMK  000537
MC$XBS  001751          MD.DOB  000015          ME.PRI  177775          MO$MIR  000031          MP$CMX  002153
MC$XCJ  000322          MD.DUM  000002          ME.PTY  177772          MO$NAM  000001          MP$CND  000310
MC$XCR  002260          MD.FAI  000014          ME.PVA  177760          MO$NIC  000023          MP$CNU  001753
MC$XCS  002261          MD.FNA  000001          ME.RES  177761          MO$NOD  000000          MP$COB  000311
MC$XDR  001762          MD.FOB  000006          ME.ROO  177754          MO$OFF  000001          MP$CON  002126
MC$XDS  001763          MD.LOA  000001          ME.SIZ  177774          MO$ON   000000          MP$COS  001604
MC$XFR  002272          MD.LSH  000020          ME.SYS  177746          MO$OPT  000001          MP$CPF  000230
MC$XFS  002273          MD.NOB  000012          ME.UCO  177770          MO$PER  000200          MP$CPL  000232
```

```
                                                                  K 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-4
```

```
                                                                        J  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 9
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

T.NDM     000066           8-140      8-141
T.QRAC    000060           8-133      8-136
T.QRIN    000062           8-135      8-138
T.QRPD    000064           8-134      8-137
T.SITC    000072           8-131
T.TOD     000070           8-139      8-142
US.MDM  = 000020           13-408
US.MNT  = 000100           13-408
U.CW3     000014           13-416
U.STS     000005           13-408
XMTTBL    000144 R        #8-132      22-1024
X$$MCB  = ******           5-72       5-72
ZF.COU  = 001000          #5-72
ZF.DDM  = 000001          #5-72
ZF.DIA  = 004000          #5-72
ZF.DLC  = 000002          #5-72
ZF.DVP  = 100000          #5-72
ZF.INI  = 040000          #5-72
ZF.KMX  = 000020          #5-72
ZF.LLC  = 000004          #5-72
ZF.LMC  = 000100          #5-72
ZF.MAN  = 020000          #5-72
ZF.MFL  = 000010          #5-72
ZF.MTM  = 000400          #5-72
ZF.MUX  = 000040          #5-72
ZF.PSE  = 002000          #5-72
ZF.SLI  = 010000          #5-72
ZF.TIM  = 000200          #5-72
ZF.X3P  = 000000          #5-72
ZS.ASN  = 100000          #5-72       13-428
ZS.BSY  = 140000          #5-72
Z.AVL     000014          #5-72
Z.DAT     000016          #5-72       20-903
Z.DSP     000000          #5-72       5-72
Z.FLG     000010          #5-72
Z.LEN   = 000016          #5-72
Z.LLN     000006          #5-72       13-422
Z.MAP     000020          #5-72       13-424
Z.NAM     000004          #5-72       13-384     14-556     21-992
Z.PCB     000012          #5-72
Z.SCH     000007          #5-72
$DMPST  = ******   GX      23-1102
$HEADR  = ******   GX      12-301     18-751     19-802
$TMLUN  = ******   GX      13-388     13-404
$$$L    = 000023          #7-98       7-98      #7-99       7-99      #7-101      7-101
```

```
                                                                        K  8
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14      PAGE 10
```

```
                                              .SBTTL  SETSVC - Set SVC parameters
                                      ;+
                                      ; *** SETSVC - Set SVC parameters
                                      ;
                                      ; INPUTS
                                      ;       R4 - Address of context area
                                      ;
                                      ; OUTPUTS
                                      ;       R0 - Error code
                                      ;
                                      ;-

000400                                SETSVC:
000400  032764  176411  000076                BIT     #^C<LS$DLM!LS$SVC>,L$FLX(R4) ; Check only SVC DLM parameters
000406  001045                                BNE     30$                         ; Error if trying to set something else

000410                                        SWSTK$  40$                         ;; Do SVC processing in system state
000414                                        SAVMAP                              ;; Save current mapping

000420                                        CALL    CKCSTA                      ;; Check circuit is off
000424  005700                                TST     R0                          ;; Check error status
000426  100430                                BMI     10$                         ;; Branch if circuit in wrong state
000430                                        CALL    FNDCTR                      ;; Find XPT counters
000434  103420                                BCS     5$                          ;; Error if counters do not exist
000436                                        CALL    SETCTR                      ;; Set counter block flags
000442  016105  000036                        MOV     T$SVC+2(R1),R5              ;; Get virtual SVC block address
000446                                        MAP     T$SVC(R1)                   ;; Map to SVC descriptor
000454                                        CALL    SVCFLG                      ;; Set SVC flags
000460  016401  000076                        MOV     L$FLX(R4),R1                ;; Get remaining parameters to set
000464  012703  000000'                       MOV     #SVCMAP,R3                  ;; Point to parameter descriptions
000470                                        CALL    $STPAR                      ;; Set remaining parameters
000474  000407                                BR      20$                         ;; Exit

000476  012700  177765                5$:     MOV     #ME.CST,R0                  ;; Return wrong state if no counters
000502  012764  000003  000072                MOV     #ME$CIR,L$PAR(R4)           ;; ... this should be obvious
000510  010066  000004                10$:    MOV     R0,R$R0+2(SP)               ;; Return error to caller

000514                                20$:    RESMAP                              ;; Restore mapping
000520                                        RETURN

000522  012700  177745                30$:    MOV     #ME.GRO,R0                  ; Parameter grouping error
000526  012764  177777  000072                MOV     #-1,L$PAR(R4)               ; No further information

000534                                40$:    RETURN
```

```
                                                                                 J 10
CICHPX      CREATED BY  MACRO  ON 29-JUN-85 AT 12:15       PAGE 2
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE             REFERENCES

LN.OAU  = 000003          #5-72
LN.OFF  = 000001          #5-72       11-272      11-283
LN.ON   = 000000          #5-72
LN.OOP  = 000004          #5-72
LN.OPE  = 000001          #5-72
LN.REF  = 000002          #5-72
LN.SER  = 000002          #5-72
LN.STA  = 000017          #5-72
LN.SUB  = 000360          #5-72
LN.TRI  = 000006          #5-72
LP$MPT  = 000010          #6-104      11-265
LP$MUX  = 000004          #6-104
LP$NXC  = 100000          #6-104
LP$TRB  = 000002          #6-104      11-278
LP$UNT  = 000001          #6-104
LP$WCN  = 000040          #6-104      6-104
LP$WDV  = 000020          #6-104      6-104
LP$WLD  = 000360          #6-104
LP$WTR  = 000200          #6-104      6-104
LP$WUN  = 000100          #6-104      6-104
LS$ACT  = 000040          #6-104
LS$BBT  = 010000          #6-104
LS$BLK  = 001000          #6-104      6-104       8-160       12-313      13-340
LS$BSA  = 000001          #6-104
LS$BSD  = 000002          #6-104
LS$BSI  = 000004          #6-104
LS$CHN  = 000001          #6-104      6-104
LS$CMB  = 000002          #6-104      6-104       6-104       6-104       7-123
LS$COS  = 000004          #6-104
LS$CUS  = 000004          #6-104      6-104       8-155       13-334
LS$DDT  = 001000          #6-104
LS$DEA  = 000100          #6-104
LS$DLM  = 001004          #6-104      10-214
LS$DLT  = 002000          #6-104
LS$DTE  = 000010          #6-104      6-104
LS$HBT  = 000400          #6-104
LS$HTM  = 010000          #6-104
LS$INA  = 000010          #6-104
LS$IND  = 000020          #6-104
LS$INI  = 000040          #6-104
LS$LCT  = 020000          #6-104      6-104       9-190       9-193
LS$LMB  = 000002          #6-104
LS$LOO  = 040000          #6-104
LS$MDT  = 020000          #6-104
LS$MRT  = 000020          #6-104
LS$MWN  = 000040          #6-104
LS$MXB  = 001000          #6-104
LS$MXR  = 000020          #6-104      6-104       6-104       7-126       7-127
LS$MXW  = 000040          #6-104      6-104       6-104       6-104       7-124
LS$NMT  = 020000          #6-104
LS$NOR  = 100000          #6-104
LS$NTI  = 000200          #6-104
```

```
                                                                                 K 10
CICHPX      CREATED BY  MACRO  ON 29-JUN-85 AT 12:15       PAGE 3
```

```
                                                .SBTTL  CIRECH - Read circuit characteristics
                                        ;+
                                        ; **-CIRECH-READ CIRCUIT CHARACTERISTICS
                                        ;
                                        ; THIS ROUTINE IS CALLED TO RETURN INFORMATION ABOUT A CIRCUIT'S
                                        ; CHARACTERISTICS.  PARAMETERS RETURNED INCLUDE COST, TYPE, TRIBUTARY NUMBER,
                                        ; OWNER AND MULTIPOINT POLLING PARAMETERS.
                                        ;
                                        ; INPUTS:
                                        ;       R3 = ADDRESS OF NEXT FREE BYTE IN OUTPUT BUFFER
                                        ;       R4 = ADDRESS OF CONTEXT AREA
                                        ;
                                        ; OUTPUTS:
                                        ;       R3 = ADDRESS OF NEXT FREE BYTE IN OUTPUT BUFFER
                                        ;
                                        ; REGISTERS:
                                        ;       R4 IS PRESERVED
                                        ;-

000022                                  CIRECH::
000022  016400  000042                          MOV     L$SLT(R4),R0            ; GET THE SLT POINTER
000026  016401  000050                          MOV     L$TPT(R4),R1            ; GET THE TRIB FLAGS POINTER
000032  005067  177762                          CLR     FLAG                    ; INITIALIZE LOCAL FLAGS WORD
000036  022701  177777                          CMP     #-1,R1                  ; IS THIS A PVC?
000042  001002                                  BNE     15$                     ; BR IF NO

                                                .IF NDF R$$PRO  ; PRO/DECNET

000044                                          CALLR   PVCCHA                  ; GET PVC CHARACTERISTICS

                                                .IFF    ; NDF R$$PRO

                                                RETURN                          ; RETURN

                                                .ENDC   ; NDF R$$PRO

                                        ;
                                        ; Return service, adjacent node, designated router, cost.
                                        ;
000050  032760  040000  000000          15$:    BIT     #LF.RDY,L.FLG(R0)       ; Is line loaded?
000056  001007                                  BNE     17$                     ; if NE, ok
000060  042764  100000  000064                  BIC     #LF$SEG,L$FLG(R4)       ; else avoid infinite loop,

000066                                          SWSTK$  45$                     ;; get counter block for DLM usage
000072                                          CALLR   FNDCIR                  ;; and return to user state at 45$

000076  032764  040000  000064          17$:    BIT     #LF$MLT,L$FLG(R4)       ; processing multiple adjacencies ?
000104  001002                                  BNE     18$                     ; Service not qualified by adjacent node
000106                                          CALL    SETSER                  ; Get service parameter

000112  005067  177676                  18$:    CLR     LTIM                    ; assume no listen timer/adjacent node
000116                                          CALL    FNDLTM                  ; get adjacent node info
000122  103405                                  BCS     20$                     ; if CS, no adjacency
000124  016702  177650                          MOV     NODE,R2                 ; get node address
000130  004567  002132                          JSR     R5,SETNOD               ; stuff it into message
000134  001440                                  .WORD   MP$ADJ                  ; parameter = adjacent node
000136  032764  040000  000064          20$:    BIT     #LF$MLT,L$FLG(R4)       ; processing multiple adjacencies ?
```

```
                                             .SBTTL  FNDCTR -  Find counter block
                                     ;+
                                     ;
                                     ; FNDCTR - FIND TRANSPORT LINE COUNTER BLOCK
                                     ;
                                     ; INPUTS:
                                     ;          R4 - ADDRESS OF CONTEXT AREA
                                     ;
                                     ; OUTPUTS:
                                     ;          CARRY CLEAR:
                                     ;               R1 - ADDRESS OF FOUND COUNTER BLOCK
                                     ;               FLAG - INDICATES IF CIRCUIT IS DLM SVC OR PVC
                                     ;               SVCMAP - MAPPING BIAS FOR SVC DESCRIPTOR IF SVC
                                     ;               SVCADD - VIRTUAL ADDRESS OF SVC DESCRIPTOR IF SVC
                                     ;               PVCPRT - PVC PORT NUMBER IF PVC
                                     ;          CARRY SET:
                                     ;               COUNTER BLOCK NOT FOUND
                                     ;
                                     ; REGISTERS:    R0, R1 AND R5 ARE CHANGED.
                                     ;
                                     ; *** NOTE: THIS ROUTINE MUST BE CALLED FROM SYSTEM STATE
                                     ;           APR6 IS MODIFIED
                                     ;
                                     ;-
002432  016400  000042               FNDCTR: MOV     L$SLT(R4),R0            ;; GET THE SLT ADDRESS
002436  116001  000021                       MOVB    L.OWNR(R0),R1           ;; ASSUME OWNER = LINE OWNER
002442  032760  000400  000000               BIT     #LF.BRO,L.FLG(R0)       ;; BROADCAST CIRCUIT?
002450  001007                               BNE     5$                      ;; BR IF YES
002452  032760  000020  000000               BIT     #LF.MTP,L.FLG(R0)       ;; MULTIPOINT LINE ?
002460  001407                               BEQ     10$                     ;; IF EQ, NO
002462  105760  000014                       TSTB    L.NSTA(R0)              ;; MULTIPOINT TRIB (SLAVE)?
002466  001404                               BEQ     10$                     ;; BR IF YES
002470  016401  000050               5$:     MOV     L$TPT(R4),R1            ;; GET THE TRIB FLAGS POINTER
002474  116101  000003                       MOVB    S.OWNR(R1),R1           ;; ELSE, GET TRIB OWNER
002500  067701  000000G              10$:    ADD     @PDVTA,R1               ;; POINT TO PDV TABLE ENTRY     ; RJK10
002504  011101                               MOV     (R1),R1                 ;; GET THE PDV ADDRESS
002506  022761  114224  000004               CMP     #^RXPT,Z.NAM(R1)        ;; IS THE OWNER TRANSPORT ?
002514  001132                               BNE     40$                     ;; IF NE, NO-PARAMETER NOT APPLICABLE
002516  005046                               CLR     -(SP)                   ;; GET SLN NUMBER
002520  017705  000000G                      MOV     @SLTMA,R5               ;; POINT TO SLT VECTOR TABLE    ; RJK10
002524  020025                       15$:    CMP     R0,(R5)+                ;; IS THIS THE CORRECT SLT?
002526  001402                               BEQ     17$                     ;; BR IF YES
002530  005216                               INC     (SP)                    ;; ELSE UPDATE SLN
002532  000774                               BR      15$                     ;; AND CHECK NEXT
002534  116466  000040  000001       17$:    MOVB    L$TRB(R4),1(SP)         ;; SET TRIB NUMBER ONTO HIGH BYTE
002542  012605                               MOV     (SP)+,R5                ;; RETRIEVE SLN/STA
002544                                       MAP     Z.DSP(R1)               ;; MAP TO XPT'S PROCESS SPACE
002552  016101  000016                       MOV     Z.DAT(R1),R1            ;; POINT TO XPT'S DDB
002556  016100  000100                       MOV     N$TLC(R1),R0            ;; GET THE NUMBER OF LINE COUNTERS

                                             .IF NDF X$$TLC

                                             MOV     N$TLC+2(R1),R1          ;; POINT TO THE FIRST

                                             .IFF    ; NDF X$$TLC

002562                                       MAP     N$TLC+2(R1)             ;; MAP TO IT
```

```
                                                                J 13
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 2
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

E$NLEN    000050         #6-63
E$NLLA    000012         #6-63
E$NLNK    000000         #6-63
E$NML     000040         #6-63
E$NMR     000024         #6-63
E$NMS     000030         #6-63
E$NNOD    000002         #6-63
E$NRT     000042         #6-63
E$NRTP    000005         #6-63
E$NSEG    000010         #6-63
E$NTIM    000046         #6-63
E$NUSE    000004         #6-63
E$STRT    000006         #6-63
FLAG      000020 R       #8-97      *9-125     9-235      9-265      9-268      9-305      *17-777    *17-789    *17-794
                         *18-832
FL.INC  = 000004         #8-100      9-268     17-789     18-832
FL.PVC  = 000002         #8-99       17-794
FL.SVC  = 000001         #8-98       9-235     9-265      9-305      17-777
FMTOWN    001340 R        9-257     #12-442
FNDCTR    002432 R        9-147      13-474    #17-728
FNDDCP  = ******   GX     11-386     11-411
FNDHTM    001402 R        9-208     #13-471
FNDLTM    001474 R        9-154     #14-507
FNDPCL  = ******   GX     11-390
FNDPLD    001064 R        9-162      9-194     #10-348     14-508
GETADJ    002152 R        14-549     14-565     14-572    #15-622
HTIM      000012 R       #8-94       9-213      9-214     *13-478
I$$AS   = ******          6-69       14-588
KISAR6  = ******   GX     11-385    *11-397     11-410    *11-418     13-473    *13-483    *14-530     15-623    *15-624
                         *15-639    *15-643    *17-750    *17-760     17-783    *17-785    *17-791     18-818    *18-819
                         *18-833     19-854    *19-855    *19-883
LF$END  = 000020         #7-81       9-198      14-525
LF$MLT  = 040000         #7-81       9-149      9-159      9-223      9-241      9-329      9-332
LF$REA  = 000001         #7-81
LF$SEG  = 100000         #7-81       9-144      9-330      14-558     14-574     14-579
LF$SIG  = 000040         #7-81
LF$SKP  = 000004         #7-81
LF$VR2  = 000010         #7-81
LF$ZER  = 000002         #7-81
LF.ACT  = 100000         #6-70
LF.BRO  = 000400         #6-70       9-287      17-730
LF.BWT  = 000007         #6-70
LF.ENA  = 002000         #6-70
LF.LPB  = 001000         #6-70
LF.MDC  = 000100         #6-70
LF.MFL  = 004000         #6-70
LF.MTP  = 000020         #6-70       9-290      9-320      17-732
LF.PAC  = 000200         #6-70
LF.RDY  = 040000         #6-70       9-142      9-180
LF.REA  = 010000         #6-70
LF.SER  = 000040         #6-70       20-908
LF.TIM  = 000010         #6-70
```

K 13

```
     67                                 .SBTTL  CONTEXT AREA DEFINITIONS
     68
     69 000000                          CIRCX$  LIST
                                ;
                                ; CONTEXT AREA OFFSET DEFINITIONS
                                ;
        000020                          .ASECT
                000000                  . = 0
                000020                  L$NLEN  = 16.           ; MAXIMUM NAME LENGTH
                                ;
                                ; The offsets L$NAM to L$PDV inclusive must appear in the same order
                                ; as in the CIRCX$ and LOCCX$ macros.
                                ;
        000000                  L$NAM:
        000000                  L$DDM:  .BLKW   1               ; DEVICE NAME
        000002                  L$CTL:  .BLKB   1               ; CONTROLLER NUMBER
        000003                  L$UNT:  .BLKB   1               ; UNIT NUMBER
        000004                  L$PVC:  .BLKW   1               ; 3RD WORD OF PVC NAME
                000020                  . = 0+L$NLEN    ; SPACE FOR EXPANDED CIRCUIT NAME
        000020                  L$SCN:  .BLKB   L$NLEN          ; CURRENT NAME IN WILDCARD SCAN
        000040                  L$TRB:  .BLKB   1                       ; TRIBUTARY NUMBER
        000041                  L$PFG:  .BLKB   1                       ; PARSE FLAGS
        000042                  L$SLT:  .BLKW   1                       ; Current SLT/PVC address
        000044                  L$NXT:  .BLKW   1                       ; Pointer to next SLT/PVC address
        000046                  L$SNM:  .BLKW   1                       ; REMAINING NUMBER OF SYSTEM LINES
        000050                  L$TPT:  .BLKW   1                       ; Current tributary pointer (-1 for PSI)
        000052                  L$TNM:  .BLKB   1                       ; REMAINING NUMBER OF TRIBUTARIES
        000053                  L$CTB:  .BLKB   1                       ; CURRENT TRIBUTARY NUMBER
        000054                  L$CHN:  .BLKB   1                       ; Channel number / X.25 port number and
        000055                  L$PDV:  .BLKB   1                       ; ... PDV assigned
        000056                  L$MSG:  .BLKW   1                       ; ERROR MESSAGE STRING POINTER
        000060                  L$BUF:  .BLKW   1                       ; SAVED BUFFER POINTER
        000062                  L$OPT:  .BLKB   1                       ; SAVED OPTIONS BYTE
        000063                  L$TYP:  .BLKB   1                       ; LINE-ID FORMAT TYPE
        000064                  L$FLG:  .BLKW   1                       ; FLAG WORD
        000066                  L$SLEN: .BLKB   1                       ; Significant length of circuit name
        000067                  L$PRO:                                  ; Line protocol
        000067                  L$MTYP: .BLKB   1                       ; Network management circuit type
        000070                  L$TSZ:                                  ; Block size for transport circuits
        000070                  L$LTM:                                  ; Listen timer for transport circuits
        000070                  L$FLX:  .BLKW   1                       ; Flags word for X.25 circuit commands
        000072                  L$SCR:  .BLKW   7                       ; SCRATCH BUFFER
        000110                  L$ADJ:  .BLKW   1                       ; Pointer to adjacency database
        000112                  L$NOD:  .BLKW   1                       ; Adjacent node address
        000114                  L$COU:  .BLKW   1                       ; Count of adjacency entries
        000116                  L$PLB:  .BLKW   1                       ; PLB address
        000120                  L$PAR:  .BLKW   1                       ; PARAMETER TYPE
        000122                  L$SYL:  .BLKW   1                       ; System line number
        000124                  L$LEN:                                  ; LENGTH OF CONTEXT AREA
        000000                          .PSECT
                                ;
                                ; OFFSETS INTO SCRATCH BUFFER (L$SCR) FOR DMC BASE TABLE COUNTERS
                                ;
        000124                          .ASECT
                000000                  .=0
        000000                  B$5:    .BLKB   1                       ; BASE TABLE + 5
        000001                  B$6:    .BLKB   1                       ; BASE TABLE + 6
```

```
                                                                  J 15
CIRECO     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17      PAGE 3
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE              REFERENCES

L$SNM      000046           #6-69
L$SYL      000122           #6-69
L$TNM      000052           #6-69
L$TPT      000050           #6-69       10-142
L$TRB      000040           #6-69
L$TSZ      000070           #6-69
L$TYP      000063           #6-69
L$UNT      000003           #6-69
L.COST     000015           #5-63
L.CTL      000012           #5-63
L.CVA      177776           #5-63
L.DDM      000002           #5-63
L.DDS      000004           #5-63
L.DLC      000003           #5-63       10-151
L.DLM      000006           #5-63
L.DLS      000010           #5-63
L.FLG      000000           #5-63       10-149
L.KRBA     000016           #5-63
L.LEN    = 000022           #5-63
L.MPF      000022           #5-63
L.NMST     000020           #5-63
L.NSTA     000014           #5-63
L.OWNR     000021           #5-63
L.UNT      000013           #5-63
ME$CIR     000003             10-169
ME.UCO     177770             10-168
PCLCIR   = ******   GX        9-103
PDVTA    = ******   GX        10-152
PVCCOU   = ******   GX        10-144
R$$PRO   = ******              9-97     10-140
R$$RTR   = ******              9-93
R$$11D   = ******              5-64
R$$11M   = 000000              5-64
R$$11S   = ******              5-64     9-88         10-140
SCRATC     000000 R         #8-78
SF.ACT   = 000200           #5-63
SF.ENA   = 000100           #5-63
SF.LPB   = 000004           #5-63
SF.MFL   = 000040           #5-63
SF.PAC   = 000020           #5-63
SF.REA   = 000010           #5-63
SF.SER   = 000001           #5-63
SF.SVC   = 000002           #5-63
SF.UNL   = 000040           #5-63
S$$BAS   = ******              9-93
S.COST     000001           #5-63
S.FLG      000000           #5-63
S.LEN      000004           #5-63
S.NMST     000002           #5-63
S.OWNR     000003           #5-63
X$$MCB   = ******              5-64     5-64
ZF.COU   = 001000           #5-64       10-154
```

```
                                                                  K 15
```

J 16

```
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17  Page 6-6
Symbol table


Elapsed time: 00:00:26.49
SY:CIREIN.V2,[135,134]CIREIN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CIREIN
```

K 16

```
                                                                  K  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 5
SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL  VALUE            REFERENCES

L$BLK     000262         #6-72
L$BSA     000122         #6-72       *32-1143
L$BSD     000123         #6-72       *33-1161
L$BSI     000124         #6-72       *34-1179
L$BUF     000060         #6-72       *10-534      10-571      10-580
L$CHN     000054         #6-72
L$CMB     000234         #6-72       *21-906     *21-907
L$COS     000101         #6-72       *11-612
L$CTB     000053         #6-72
L$CTL     000002         #6-72
L$CUS     000260         #6-72
L$DDM     000000         #6-72        14-723      19-856      19-858
L$DDT     000106         #6-72
L$DEA     000105         #6-72
L$DLT     000110         #6-72
L$DTE     000240         #6-72        22-930
L$DTEL    000236         #6-72       *22-928
L$DTEP    000250         #6-72
L$FLG     000064         #6-72       *10-574
L$FLX     000076         #6-72
L$FL1     000066         #6-72
L$FL2     000070         #6-72
L$HBT     000254         #6-72
L$HTM     000226         #6-72       *18-833     *18-834
L$INA     000125         #6-72       *35-1197
L$IND     000126         #6-72       *36-1215
L$INI     000127         #6-72       *37-1233
L$LCT     000224         #6-72       *12-631     *12-632
L$LEN     000264         #6-72
L$LMB     000234         #6-72
L$LTM     000230         #6-72
L$MRT     000253         #6-72
L$MSG     000056         #6-72
L$MTYP    000075         #6-72
L$MWN     000252         #6-72
L$MXB     000133         #6-72       *41-1305
L$MXR     000253         #6-72       *24-977
L$MXW     000252         #6-72       *25-994
L$NAM     000000         #6-72
L$NLEN  = 000020         #6-72        6-72        6-72
L$NMT     000116         #6-72       *30-1107
L$NTI     000256         #6-72
L$NTL     000134         #6-72
L$NUM     000240         #6-72        23-956
L$NUML    000236         #6-72       *23-954
L$NXT     000044         #6-72
L$OPT     000062         #6-72       *10-533      10-539      10-540      10-557
L$OWN     000102         #6-72       *13-691
L$PAR     000072         #6-72       *10-545     *10-554      10-575     *15-758     *15-763     *16-788
L$PDV     000055         #6-72
L$PFG     000041         #6-72        14-719      17-811
L$PLT     000112         #6-72       *28-1071

                                                                  L  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 6
```

```
        000125                          L$INA:  .BLKB   1               ; DMP ACTIVE INCREMENT
        000126                          L$IND:  .BLKB   1               ; DMP DYING INCREMENT
        000127                          L$INI:  .BLKB   1               ; DMP INACTIVE INCREMENT
        000130                          L$TH1:  .BLKB   1               ; DMP DEAD THRESHOLD
        000131                          L$TH2:  .BLKB   1               ; DMP DYING THRESHOLD
        000132                          L$TH3:  .BLKB   1               ; DMP INACTIVE THRESHOLD
        000133                          L$MXB:  .BLKB   1               ; DMP MAXIMUM BLOCKS
        000134                          L$NTL:  .BLKW   13.             ; NTL MESSAGE BUFFER
        000166                          L$SCR:  .BLKW   15.             ; SCRATCH BUFFER
        000224                          L$LCT:  .BLKW   1               ; COUNTER TIMER
        000226                          L$HTM:  .BLKW   1               ; HELLO TIMER
        000230                          L$SER:                          ; SERVICE
        000230                          L$LTM:  .BLKW   1               ; LISTEN TIMER
        000232                          L$XCH:  .BLKW   1               ; X25 Logical Channel Number (LCN)
        000234                          L$LMB:                          ; X25 Max Block (Line)
        000234                          L$CMB:  .BLKW   1               ; X25 Max Data (Circuit)
        000236                          L$NUML:                         ; DLM Number length
        000236                          L$DTEL: .BLKW   1               ; X25 DTE length (Circuit)
        000240                          L$NUM:                          ; DLM Number
        000240                          L$DTE:  .BLKB   8.              ; X25 DTE (Circuit)
        000250                          L$DTEP: .BLKW   1               ; X25 Pointer to DTE descriptor (Circuit)
        000252                          L$MWN:                          ; X25 Max Window (Line)
        000252                          L$MXW:  .BLKB   1               ; X25 Max Window (Circuit)
        000253                          L$MRT:                          ; X25 Max Retransmits (Line)
        000253                          L$MXR:  .BLKB   1               ; DLM Max Recalls (Circuit)
        000254                          L$HBT:  .BLKW   1               ; X25 Holdback Timer (Line)
        000256                          L$NTI:                          ; X25 Retransmit Timer (Line)
        000256                          L$RET:  .BLKW   1               ; DLM Recall Timer (Circuit)
        000260                          L$CUS:  .BLKW   1               ; DLM Usage
        000262                          L$BLK:  .BLKW   1               ; DLM blocking state
                                                .IF DF  R$$PRO  ;PRO/DECnet
                                        L$LTY:  .BLKW   1               ; Loopback Type
                                        L$MDT:  .BLKW   1               ; Modem Test
                                                .ENDC   ; DF R$$PRO
                                                .EVEN
        000264                          L$LEN:                          ; LENGTH OF CONTEXT AREA
        000000                                  .PSECT

                                        ;
                                        ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                        ;
                000001                          LF$REA = 1              ; READ COUNTERS OPERATION
                000002                          LF$ZER = 2              ; ZERO COUNTERS OPERATION
                000004                          LF$SKP = 4              ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
                000010                          LF$VR2 = 10             ; CONNECTED TO VERSION 2.0 NCP
                040000                          LF$MLT  = 40000         ; MULTIPLE ADJACENCY FLAG
                100000                          LF$SEG  = 100000        ; SEGMENTED RESPONSE IN PROGRESS

                                        ;
                                        ; PARSE FLAG DEFINITIONS (L$PFG)
                                        ;
                000001                          LP$UNT = 1              ; UNIT NUMBER FOUND
                000002                          LP$TRB = 2              ; TRIBUTARY NUMBER FOUND
                000004                          LP$MUX = 4              ; DEVICE IS MUX
                000010                          LP$MPT = 10             ; LINE IS MULTIPOINT
```

```
                                                                    K 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-3
Symbol table

L.UNT   000013    MC$NOR  000000    MD$DMC  000014    ME$CIR  000003    MF$TES  000022
MB$DIS  000001    MC$NPF  001616    MD$DMF  000046    ME$EXA  000000    MF$TRA  000301
MB$ENA  000000    MC$NPR  001630    MD$DMP  000022    ME$EXE  000200    MF$TRI  000021
MB$FUL  000002    MC$NRE  001200    MD$DMR  000050    ME$LIN  000001    MF$ZER  000025
MB$MIX  000002    MC$NRT  001166    MD$DMV  000042    ME$LOG  000002    ML$ALL  100000
MB$ONE  000001    MC$NVR  001642    MD$DN   000016    ME$MOD  000004    ML$CLS  000000
MB$RXO  000001    MC$P08  000000    MD$DP   000000    ME$NOD  000000    ML$CON  000001
MB$TXO  000000    MC$P11  000001    MD$DPV  000044    ME$NON  177777    ML$EXT  000001
MB$ZER  000000    MC$RFL  002046    MD$DQ   000006    ME$OBJ  000007    ML$FIL  000002
MC$BAB  006073    MC$SBU  002051    MD$DTE  000024    ME$OB2  000004    ML$FIR  000000
MC$BID  001765    MC$SFL  002044    MD$DU   000002    ME$PRO  000005    ML$INT  000000
MC$BMC  001767    MC$SYC  000012    MD$DUP  000012    ME$SYS  000006    ML$KNO  140000
MC$BSC  001766    MC$SYL  000036    MD$DV   000026    ME.BLO  177744    ML$MON  000003
MC$CAC  001442    MC$SYR  000050    MD$DZ   000030    ME.CON  177753    ML$SYS  000002
MC$CAP  001440    MC$SYS  000024    MD$FUL  000000    ME.CST  177765    ML$TOP  000001
MC$CCL  001445    MC$SYZ  000000    MD$HAL  000001    ME.DIS  177755    MN$UNL  000377
MC$CDC  002045    MC$TBR  001140    MD$HEL  000002    ME.DON  177600    MO$ACC  000200
MC$CDP  001441    MC$TBS  001141    MD$KCP  000013    ME.FCO  177762    MO$ADD  000002
MC$CIF  001465    MC$TDR  002053    MD$KDP  000034    ME.FIO  177756    MO$ALA  000004
MC$CLD  001464    MC$TIM  000000    MD$KDZ  000036    ME.FOP  177763    MO$ALI  000000
MC$COU  100000    MC$TYP  007777    MD$KL   000040    ME.FOR  177776    MO$CHA  000040
MC$CTL  001454    MC$UBU  002052    MD$KMX  000054    ME.FUN  177777    MO$CIR  000003
MC$CTR  001452    MC$UFD  002047    MD$KMY  000052    ME.GRO  177745    MO$CLE  000100
MC$CTS  001453    MC$UMR  001132    MD$LEN  000050    ME.HAR  177750    MO$COU  000060
MC$DOV  002050    MC$UMS  001133    MD$PCL  000011    ME.IID  177767    MO$DAC  000000
MC$LBR  001750    MC$VAX  000003    MD$QNA  000005    ME.LCO  177766    MO$DEF  000000
MC$LBS  001751    MC$WID  060000    MD$UNA  000001    ME.LPR  177757    MO$DPR  000000
MC$LDI  001774    MC$WIH  040000    MD$WIT  000002    ME.MPR  177773    MO$ENT  000017
MC$LDO  001775    MC$WIL  020000    MD.ABO  000017    ME.MVE  177771    MO$EVE  000100
MC$LDR  001762    MC$W08  020000    MD.ACC  000010    ME.OPE  177747    MO$INF  000160
MC$LDS  001763    MC$W16  040000    MD.AOB  000016    ME.PLO  177751    MO$INS  000002
MC$LLB  002021    MC$W32  060000    MD.BOB  000011    ME.PMI  177743    MO$LIN  000001
MC$LLP  002115    MC$XBR  001750    MD.DIA  000007    ME.PNA  177752    MO$LOG  000001
MC$LLR  002007    MC$XBS  001751    MD.DOB  000015    ME.PRI  177775    MO$MIR  000031
MC$LOO  000001    MC$XCJ  000322    MD.DUM  000002    ME.PTY  177772    MO$NAM  000001
MC$LRB  002020    MC$XCR  002260    MD.FAI  000014    ME.PVA  177760    MO$NIC  000023
MC$LRP  002114    MC$XCS  002261    MD.FNA  000001    ME.RES  177761    MO$NOD  000000
MC$LRR  002006    MC$XDR  001762    MD.FOB  000006    ME.ROO  177754    MO$OFF  000001
MC$LSI  002032    MC$XDS  001763    MD.LOA  000001    ME.SIZ  177774    MO$ON   000000
MC$LST  002033    MC$XFR  002272    MD.LSH  000020    ME.SYS  177746    MO$OPT  000001
MC$LUP  006072    MC$XFS  002273    MD.NOB  000012    ME.UCO  177770    MO$PER  000200
MC$MAP  010000    MC$XLJ  000323    MD.NON  177777    MF$ACT  177776    MO$PRO  000002
MC$MBL  001764    MC$XLR  002330    MD.PER  000000    MF$ADD  000000    MO$PRS  000001
MC$MBX  004420    MC$XMA  000310    MD.RES  000004    MF$ADJ  177774    MO$REA  000200
MC$MBY  001752    MC$XMC  002305    MD.ROB  000005    MF$ALL  177775    MO$SET  000000
MC$MPX  004406    MC$XMS  002304    MD.RSH  000013    MF$BYE  000300    MO$STA  000020
MC$NAP  001604    MC$XNR  002332    MD.SDU  000005    MF$CHA  000023    MO$SUM  000000
MC$NBR  001130    MC$XRC  002316    MD.SLO  000003    MF$DUM  000020    MO$TSK  000000
MC$NBS  001131    MC$XRR  002331    MD.SNA  000000    MF$EVT  000001    MO$VOL  000000
MC$NCR  001154    MC$XRS  002342    MD.TLO  000004    MF$FUP  000264    MO$ZER  000000
MC$NCS  001155    MC$020  000002    MD.UNA  000002    MF$KNO  177777    MP$ACB  000012
MC$NML  001274    MD$CI   000007    MD.UNR  000003    MF$LOA  000017    MP$ACC  000514
MC$NMR  001142    MD$CNA  000003    MD.UOB  000007    MF$LOO  177775    MP$ACT  004526
MC$NMS  001143    MD$COU  000001    MD.VOL  000006    MF$REA  000024    MP$ADD  000766
MC$NNO  001606    MD$DA   000010    ME$ALI  000010    MF$SIG  177773    MP$ADJ  001440
MC$NNU  001605    MD$DL   000004    ME$AL2  000003    MF$SPF  000302    MP$ADP  006325
MC$NOP  001607    MD$DLV  000020    ME$ARE  000005    MF$SYS  000026    MP$ADS  004406
```

```
                                                                    L 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-4
```

K 4

```
**FILE**ID**CICHPD




  CCCCCCCC    IIIIII      CCCCCCCC  HH      HH  PPPPPPPP    DDDDDDDD
  CCCCCCCC    IIIIII      CCCCCCCC  HH      HH  PPPPPPPP    DDDDDDDD
CC              II      CC          HH      HH  PP      PP  DD      DD
CC              II      CC          HH      HH  PP      PP  DD      DD
CC              II      CC          HH      HH  PP      PP  DD      DD
CC              II      CC          HH      HH  PP      PP  DD      DD
CC              II      CC          HHHHHHHHHH  PPPPPPPP    DD      DD
CC              II      CC          HHHHHHHHHH  PPPPPPPP    DD      DD
CC              II      CC          HH      HH  PP          DD      DD
CC              II      CC          HH      HH  PP          DD      DD
CC              II      CC          HH      HH  PP          DD      DD      ....
CC              II      CC          HH      HH  PP          DD      DD      ....
  CCCCCCCC    IIIIII      CCCCCCCC  HH      HH  PP          DDDDDDDD        ....
  CCCCCCCC    IIIIII      CCCCCCCC  HH      HH  PP          DDDDDDDD        ....




LL            SSSSSSSS  TTTTTTTTTT
LL            SSSSSSSS  TTTTTTTTTT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL          SS              TT
LL            SSSSSS        TT
LL            SSSSSS        TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LL                  SS      TT
LLLLLLLLLL  SSSSSSSS        TT
LLLLLLLLLL  SSSSSSSS        TT
```

L 4

CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13

```
    265                                         .IF NDF R$$11S
```

```
                                                                         K 6
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 18-1
LINON - TURN CIRCUIT ON

    765 002372  152760  000200  000022              BISB    #SF.ACT,L.MPF(R0)       ; MARK THE STATION ACTIVE
    766 002400  052761  100000  000000      50$:    BIS     #LF.ACT,L.FLG(R1)       ; MARK THE CIRCUIT ACTIVE
    767 002406  005727                              TST     (PC)+                   ; SET SUCCESS
    768 002410  000261                      60$:    SEC                             ; FAILURE
    769 002412                                      RETURN
```

```
                                                                         L 6
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 19
```

```
                                                                                  K  7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-4
Symbol table

MP$CPT   000226        MP$LHL   000232        MP$NXN=  001476        MP$SLO   000171        MS$OFF   000001
MP$CPU   000161        MP$LLE   000227        MP$OAC   000632        MP$SMX   002202        MS$ON    000000
MP$CSZ   001755        MP$LLD   004432        MP$OAN   000620        MP$SND   000500        MS$PRO   000006
MP$CUS   002127        MP$LMB   002152        MP$OCO   000776        MP$SNP   006332        MS$REA   000004
MP$CVA   000540        MP$LMX   002200        MP$OHO   000214        MP$SNU   000542        MS$REF   000001
MP$DAL   002570        MP$LNA   000144        MP$ONA   000764        MP$SOB   000524        MS$RES   000003
MP$DCO=  001464        MP$LNO   000233        MP$ONR   002114        MP$SPA   000157        MS$RST   000001
MP$DDT   002177        MP$LOA   000170        MP$OQL   001453        MP$SPR   000536        MS$RSX   000002
MP$DEL   001131        MP$LOG   004514        MP$OTI   000777        MP$SPS   000513        MS$RT    000005
MP$DES   000156        MP$LOO   000620        MP$OTY   001022        MP$STA   000000        MS$SER   000002
MP$DEV   002114        MP$LTM   001613        MP$OUS   000777        MP$STI   002140        MS$SHU   000002
MP$DFA   001320        MP$LTY   004533        MP$OVE   001010        MP$STT   002201        MS$SRV   000007
MP$DHO=  001465        MP$LWI   000230        MP$OWN   004374        MP$STY   000175        MS$STA   000000
MP$DIA   000173        MP$MAC   004420        MP$PAR   000036        MP$SUB   000001        MS$SYN   000012
MP$DLB   002571        MP$MAD   001630        MP$PAS   000513        MP$SUR   000156        MS$TER   000001
MP$DLG   004521        MP$MAP   006323        MP$PCH   002152        MP$SUS   000512        MS$TOP   000003
MP$DLI=  001466        MP$MAR   001635        MP$PCO   000024        MP$SVR   000163        MS$TRI   000005
MP$DLT   002200        MP$MAV   006321        MP$PCT   000144        MP$SWI   002260        MS$UNR   000005
MP$DRO   001441        MP$MBN   001636        MP$PDT   002114        MP$SYP   002424        MS$VMS   000004
MP$DST   000454        MP$MBR   001637        MP$PHA   000170        MP$TFL   000156        MS.DON   177600
MP$DTE   002140        MP$MBU   001642        MP$PHY   000012        MP$TH1   002206        MS.MOR   000002
MP$DTY=  001452        MP$MCB   000156        MP$PLI   000025        MP$TH2   002205        MS.PAR   000003
MP$DUA   000207        MP$MCO   001632        MP$PLN   002140        MP$TH3   002202        MS.SUC   000001
MP$DUC   000210        MP$MDE   004421        MP$PLO   000012        MP$TLN   000202        MT$AR4   000003
MP$DUM   000202        MP$MHO   001633        MP$PMC   004374        MP$TLO   000172        MT$ASC   000100
MP$DUP   002127        MP$MLB   000202        MP$PNT   002126        MP$TPA   004375        MT$BIL   000001
MP$DVC   002114        MP$MLK   001306        MP$PRI   004411        MP$TRI   002164        MT$BYS   000011
MP$DWE   001321        MP$MLN   001631        MP$PRO   002130        MP$TST   000144        MT$CI    000007
MP$ELT   000157        MP$MLP   006333        MP$PSS   001763        MP$TYP   002130        MT$CLE   000077
MP$ETY   001605        MP$MRB   002171        MP$PST   001762        MP$UCS   004407        MT$COD   000200
MP$EVE   000311        MP$MRO   001605        MP$RET   001631        MP$USR   001750        MT$CON   000001
MP$FNC   001752        MP$MRP   006322        MP$RFA   001323        MP$VEC   004410        MT$DMC   000004
MP$GDT   002222        MP$MRT   002153        MP$RMX   002201        MP$VER   004406        MT$ETH   000006
MP$GNM   002223        MP$MRV   006320        MP$RPA   004374        MP$WDF   002165        MT$HEX   000040
MP$GRO   002115        MP$MSB   000170        MP$RPR   001606        MP$WMX   002177        MT$LPB   000005
MP$GRP   000541        MP$MVE   000145        MP$RRT   002213        MP$XMT   002166        MT$MAX   000037
MP$GTY   002224        MP$MVI   001634        MP$RST   002212        MP$XPF   000242        MT$MUL   000100
MP$HAD   001757        MP$MVR   001751        MP$RSV   001754        MP$XPL   000244        MT$NLE   000017
MP$HBT   002142        MP$MWN   002154        MP$RSZ   001756        MP$XPT   000240        MT$NON   000001
MP$HDD   000162        MP$MXB   002172        MP$RTI   001616        MP$XXX   177777        MT$NR4   000005
MP$HTM   001612        MP$MXC   000466        MP$RTM   001611        MP$5ST   004406        MT$NTY   000060
MP$HWA   002210        MP$MXR   001630        MP$RVE   001604        MS$ACT   000001        MT$OCT   000060
MP$IAT   001322        MP$MXW   002143        MP$RVT   000156        MS$ADU   000010        MT$PHA   000002
MP$IDE   000144        MP$NAA   006335        MP$SAC   000514        MS$ALO   000007        MT$POI   000000
MP$IDP   006327        MP$NAC   001130        MP$SAD   000543        MS$ASE   000006        MT$QP2   000010
MP$IHO   000215        MP$NAP   006326        MP$SCA   000310        MS$ATR   000011        MT$ROU   000000
MP$INA   002177        MP$NCT   000240        MP$SCO   000144        MS$AUT   000000        MT$RO4   000004
MP$IND   002204        MP$NET   002114        MP$SCT   002176        MS$CLE   000003        MT$SEC   000000
MP$INI   002201        MP$NLI   000765        MP$SDU   000203        MS$DED   000004        MT$SGD   000020
MP$ITI   000776        MP$NNA   000764        MP$SDV   000160        MS$DIE   000003        MT$SYS   000002
MP$LAA   000231        MP$NOD   000500        MP$SEH   004401        MS$DUM   000004        MT$TER   000001
MP$LAN   000234        MP$NRA   006334        MP$SER   000144        MS$FAI   000013        MT$TRI   000002
MP$LAR   000202        MP$NRB   000214        MP$SET   000000        MS$GLO   000000        MT$TYP   007777
MP$LAS   000012        MP$NSA   001617        MP$SGZ   001644        MS$HOL   000002        MT$USD   000000
MP$LBS   000203        MP$NTI   002141        MP$SID   000176        MS$INA   000002        MT$X25   000003
MP$LCO   000226        MP$NUM   001642        MP$SIN   000310        MS$LOA   000003        MU$INC   000001
MP$LCT   000156        MP$NVE   001274        MP$SLI   000156        MS$LOO   000002        MU$OUT   000002
```

```
                                                                                  L  7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-5
CICHPD     CREATED BY  MACRO  ON 29-JUN-85 AT 12:14     PAGE 10

MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME      REFERENCES

ALUN$S          #5-55     13-388
BIAS            #5-56
CALL            #5-58     9-178     10-205    11-239    12-297    12-299    13-387    13-468    13-478    13-490
                  13-494  13-496    14-558    14-564    15-626    15-632    16-667    17-698    17-705    18-740
                  18-742  19-791    19-793    20-893    20-895    20-922    20-925    21-968    21-993    22-1019
                  22-1022 22-1025   22-1028   22-1031   22-1034   22-1037   22-1040   22-1043   22-1046   22-1049
                  22-1052 22-1055   23-1102
CALLR           #5-58     18-749    19-800
CALLX           #5-58
CCBDF$          #5-56     5-76
CEACC$          #5-55
CICCX$          #5-58     6-86
CTRDF$          #5-56     5-68
DHBDF$          #5-57     5-81
DIR$            #13-388   13-388   #18-755    18-755   #19-806    19-806
DMPDF$          #5-62     5-64
ECDDB$          #5-55     5-74
HDRDF$          #5-57     5-77
MANDF$          #5-53     5-69
MAP             #5-56    #5-56
MESAG$          #5-54     7-98      7-99      7-101
MOV$            #13-388   13-388    13-388    13-388
NSFDF$          #5-54     5-70
NTLDF$          #5-53     5-71
PDVDF$          #5-53     5-72
PLBDF$          #5-57     5-79
RESMAP          #5-56     12-304
RESRG           #5-53     11-260    13-431    13-451    13-479
RETC            #5-56     12-301    18-751    19-802
RETURN            9-179   10-206    11-261    12-305    12-311    13-501    14-574    15-644    16-669    17-706
                  18-752  18-769    19-803    19-827    20-926    20-929    21-1006   22-1020   22-1023   22-1026
                  22-1029 22-1032   22-1035   22-1038   22-1041   22-1044   22-1047   22-1050   22-1053   22-1056
                  23-1105 24-1148
SAVMAP          #5-56     12-298
SAVRG           #5-53     11-227    13-427    13-477
SLTDF$          #5-54     5-73
SOB               11-238  13-430    20-913
SWSTK$            12-297  18-740    19-791    20-893
TCBDF$          #5-57     5-80
UCBDF$          #5-57     5-78
WSIG$S          #5-55     18-755    19-806
XPDDB$          #5-57     5-75
```

```
**FILE**ID**CICHPX
```

```
                                        .SBTTL  CKCSTA - Check circuit is in STATE OFF
                                ;+
                                ; *** CKCSTA - Check circuit is in STATE OFF
                                ;
                                ; INPUTS
                                ;       R4 - Address of context area
                                ;
                                ; OUTPUTS
                                ;       Carry clear if STATE OFF and line loaded
                                ;       R0 - error code
                                ;       L$PAR(R4) - error detail
                                ;       R1 destroyed
                                ;-

000536                          CKCSTA:
000536  016401  000042                  MOV     L$SLT(R4),R1            ;; Get SLT pointer
000542  032761  040000  000000          BIT     #LF.RDY,L.FLG(R1)       ;; 'Line' must be loaded
000550  001433                          BEQ     20$                     ;; Branch if not loaded

000552  132764  000010  000041          BITB    #LP$MPT,L$PFG(R4)       ;; Is this a multipoint line?
000560  001011                          BNE     10$                     ;; Branch if multipoint
                                ;
                                ; Point to point 'line'
                                ;
000562  032761  100000  000000          BIT     #LF.ACT,L.FLG(R1)       ;; Is this circuit active?
000570  001023                          BNE     20$                     ;; Error if circuit is active
000572  122761  000001  000020          CMPB    #LN.OFF,L.NMST(R1)      ;; Check line is OFF
000600  001017                          BNE     20$                     ;; Branch if not OFF
000602  001426                          BEQ     50$                     ;; Branch if OFF
                                ;
                                ; Multipoint 'line'
                                ;
000604  132764  000002  000041  10$:    BITB    #LP$TRB,L$PFG(R4)       ;; Was trib specified?
000612  001415                          BEQ     30$                     ;; Branch if no trib given
000614  016401  000050                  MOV     L$TPT(R4),R1            ;; Get pointer to trib flags
000620  132761  000200  000000          BITB    #SF.ACT,S.FLG(R1)       ;; Ensure trib not active
000626  001004                          BNE     20$                     ;; Branch if active
000630  122761  000001  000002          CMPB    #LN.OFF,S.NMST(R1)      ;; Check trib is OFF
000636  001410                          BEQ     50$
                                ;
                                ; Wrong state error
                                ;
000640  012700  177765          20$:    MOV     #ME.CST,R0              ;; Return wrong state error
000644  000402                          BR      40$
                                ;
                                ; No tributary error
                                ;
000646  012700  177767          30$:    MOV     #ME.IID,R0              ;; Return invalid format error
000652  012764  000003  000072  40$:    MOV     #ME$CIR,L$PAR(R4)       ;; Indicate component in error
                                ;
                                ; Common return
                                ;
000660                          50$:    RETURN
```

```
                                                                K 10
CICHPX      CREATED BY  MACRO  ON 29-JUN-85 AT 12:15      PAGE 3
SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL  VALUE             REFERENCES

LS$NTL  = 000001          #6-104
LS$NUM  = 000100          #6-104     6-104     7-125     7-130
LS$OWN  = 000010          #6-104
LS$PLT  = 004000          #6-104
LS$PRO  = 010000          #6-104
LS$PVC  = 020053          #6-104     9-188
LS$RET  = 000200          #6-104     6-104     6-104     7-128     7-129
LS$SER  = 100000          #6-104
LS$STA  = 000002          #6-104
LS$SVC  = 000362          #6-104     10-214
LS$TAD  = 000020          #6-104
LS$TH1  = 000100          #6-104
LS$TH2  = 000200          #6-104
LS$TH3  = 000400          #6-104
LS$XMT  = 040000          #6-104
L$ACT     000104          #6-104
L$BBT     000114          #6-104
L$BLK     000262          #6-104     12-316    13-343
L$BSA     000122          #6-104
L$BSD     000123          #6-104
L$BSI     000124          #6-104
L$BUF     000060          #6-104
L$CHN     000054          #6-104
L$CMB     000234          #6-104     7-123
L$COS     000101          #6-104
L$CTB     000053          #6-104
L$CTL     000002          #6-104
L$CUS     000260          #6-104     8-157     13-337
L$DDM     000000          #6-104
L$DDT     000106          #6-104
L$DEA     000105          #6-104
L$DLT     000110          #6-104
L$DTE     000240          #6-104
L$DTEL    000236          #6-104
L$DTEP    000250          #6-104
L$FLG     000064          #6-104
L$FLX     000076          #6-104     8-155     8-160     9-188     9-190     9-193     10-214    10-229    12-313
                            13-334   13-340
L$FL1     000066          #6-104
L$FL2     000070          #6-104
L$HBT     000254          #6-104
L$HTM     000226          #6-104
L$INA     000125          #6-104
L$IND     000126          #6-104
L$INI     000127          #6-104
L$LCT     000224          #6-104     15-402
L$LEN     000264          #6-104
L$LMB     000234          #6-104
L$LTM     000230          #6-104
L$MRT     000253          #6-104
L$MSG     000056          #6-104
L$MTYP    000075          #6-104
```

```
                                                                L 10
CICHPX      CREATED BY  MACRO  ON 29-JUN-85 AT 12:15      PAGE 4
```

```
000144  001116                          BNE     35$                     ; if NE, yes - skip to listen timer

000146                                  CALL    FNDPLD                  ; locate PLB for circuit
000152  103431                          BCS     25$                     ; if CS, not a transport circuit
000154  122762  000012  000000          CMPB    #PS$UP,P$LST(R2)        ; is link state = up ?
000162  001025                          BNE     25$                     ; if EQ, no
000164  132762  000200  000001          BITB    #PT$BRO,P$TYP(R2)       ; Is this a broadcast channel ?
000172  001421                          BEQ     25$                     ; If EQ, no, not applicable
000174  005762  000036                  TST     P$DRTR(R2)              ; any endnodes on Ethernet ?
000200  001416                          BEQ     25$                     ; if EQ, no - not applicable
000202  116246  000001                  MOVB    P$TYP(R2),-(SP)         ; push link type flags
000206  016202  000036                  MOV     P$DRTR(R2),R2           ; get node address of designated router
000212  132726  000100                  BITB    #PT$DRT,(SP)+           ; are WE the designated router ?
000216  001404                          BEQ     22$                     ; if EQ, no
000220  017702  000000G                 MOV     @DECPT,R2               ; else point to DECnet home block; RJK10
000224  016202  000014                  MOV     D$LNUM(R2),R2           ; fetch exec address

000230  004567  002032          22$:    JSR     R5,SETNOD               ; stuff node id
000234  001441                          .WORD   MP$DRO                  ; designated router

000236  032760  040000  000000  25$:    BIT     #LF.RDY,L.FLG(R0)       ; Is line loaded?
000244  001414                          BEQ     31$                     ; if EQ, no - skip cost parameter
000246  004567  000000G                 JSR     R5,SETPAR               ; SET COST PARAMETER
000252  001604                          .WORD   MP$COS                  ; ...
000254  000001                          .WORD   1                       ; ONE DECIMAL BYTE

000256  116002  000015                  MOVB    L.COST(R0),R2           ; ASSUME COST = LINE COST
000262  105760  000014                  TSTB    L.NSTA(R0)              ; MULTIPOINT ?
000266  001402                          BEQ     30$                     ; IF EQ, NO - WE HAVE COST
000270  116102  000001                  MOVB    S.COST(R1),R2           ; ELSE, GET TRIB COST
000274  110223                  30$:    MOVB    R2,(R3)+                ; SET COST INTO MESSAGE
                                ;
                                ; RETURN ROUTER PRIORITY, HELLO AND LISTEN TIMERS
                                ;
000276                          31$:    CALL    FNDPLD                  ; get PLB address
000302  103424                          BCS     32$                     ; if CS, couldn't do it
000304  132762  000200  000001          BITB    #PT$BRO,P$TYP(R2)       ; is it a broadcast channel ?
000312  001420                          BEQ     32$                     ; if EQ, no
000314  032764  000020  000064          BIT     #LF$END,L$FLG(R4)       ; Is this an end node?
000322  001014                          BNE     32$                     ; If yes no max routers or router pri
000324  004567  000000G                 JSR     R5,SETPAR               ; set parameter = MAXIMUM ROUTERS
000330  001605  000001                  .WORD   MP$MRO,1!MT$USD         ; 1 byte decimal unsigned
000334  116223  000040                  MOVB    P$NRNI(R2),(R3)+        ; move it

000340  004567  000000G                 JSR     R5,SETPAR               ; else set parameter = ROUTER PRIORITY
000344  001606  000001                  .WORD   MP$RPR,1!MT$USD         ; 1 byte, decimal unsigned
000350  116223  000042                  MOVB    P$RPRI(R2),(R3)+        ; move it

000354                          32$:    CALL    FNDHTM                  ; get hello timer
000360  103410                          BCS     35$                     ; if CS, look for listen timer
000362  004567  000000G                 JSR     R5,SETPAR               ; HELLO TIMER PARAMETER
000366  001612                          .WORD   MP$HTM                  ; ...
000370  000002                          .WORD   2
000372  116723  177414                  MOVB    HTIM,(R3)+              ; STORE LOW BYTE OF HELLO TIMER
000376  116723  177411                  MOVB    HTIM+1,(R3)+            ; STORE HIGH BYTE

000402  005767  177406          35$:    TST     LTIM                    ; any listen timer found ?
```

```
002570  016101  000104                  MOV     N$TLC+4(R1),R1          ;; POINT TO THE FIRST

                                        .ENDC   ; NDF X$$TLC

002574                                  BIAS    R1                      ;; SET APR6 BIAS
002604  005300                  20$:    DEC     R0                      ;; AT END OF COUNTERS ?
002606  002475                          BLT     40$                     ;; IF LT, YES-PARAMETER NOT APPLICABLE
002610  026105  000000                  CMP     T$LIN(R1),R5            ;; ARE THESE THE COUNTERS ?
002614  001403                          BEQ     30$                     ;; IF EQ, YES - READ THEM
002616  062701  000046                  ADD     #T$LLEN,R1              ;; ELSE, POINT TO NEXT COUNTERS
002622  000770                          BR      20$                     ;; AND KEEP LOOKING
002624  000241                  30$:    CLC                             ;; INDICATE FOUND IT
002626  032761  000200  000044          BIT     #PX$DLM,T$FLAG(R1)      ;; IS THIS A DLM CIRCUIT?
002634  001463                          BEQ     50$                     ;; BR IF NO
002636  032761  000100  000044          BIT     #PX$SVC,T$FLAG(R1)      ;; IS IT AN SVC?
002644  001447                          BEQ     35$                     ;; BR IF NO
002646  012767  000001  175144          MOV     #FL.SVC,FLAG            ;; ELSE INDICATE SVC
002654  016167  000034  175122          MOV     T$SVC(R1),SVCMAP        ;; SAVE SVC APR MAPPING
002662  016167  000036  175116          MOV     T$SVC+2(R1),SVCADD      ;; SAVE VIRTUAL ADDRESS
002670  022767  120000  175110          CMP     #120000,SVCADD          ;; IS DESCRIPTOR IN DSR?
002676  101006                          BHI     33$                     ;; BR IF YES
002700                                  BIAS    SVCADD                  ;; SET APR6 BIAS
002714                          33$:    SAVMAP                          ;; save mapping
002720  010146                          MOV     R1,-(SP)                ;; save R1                      [TD002]
002722                                  MAP     SVCMAP                  ;; map to SVC descriptor
002730  016701  175052                  MOV     SVCADD,R1               ;; point to SVC descriptor
002734  032761  000002  000010          BIT     #SF$INC,S$FLG(R1)       ;; is this an incoming SVC ?
002742  001403                          BEQ     34$                     ;; if EQ, no- must be outgoing
002744  052767  000004  175046          BIS     #FL.INC,FLAG            ;; else set incoming bit in flags
002752  012601                  34$:    MOV     (SP)+,R1                ;; restore R1                   [TD002]
002754                                  RESMAP                          ;; restore mapping
002760  000241                          CLC                             ;; BE SURE CARRY IS CLEAR
002762  000410                          BR      50$                     ;; AND EXIT
002764  012767  000002  175026  35$:    MOV     #FL.PVC,FLAG            ;; INDICATE PVC
002772  116167  000036  175010          MOVB    T$SVC+2(R1),PVCPRT      ;; SAVE PVC PORT NUMBER
003000  000401                          BR      50$                     ;; AND EXIT
003002  000261                  40$:    SEC                             ;; INDICATE NOT FOUND
003004                          50$:    RETURN                          ;;
```

```
                                                                        K 13
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16        PAGE 3
SYMBOL CROSS REFERENCE                                     CREF    04.00

SYMBOL  VALUE              REFERENCES

LF.UNL  = 020000           #6-70
LF.X2P  = 000000           #6-70
LN.CLO  = 000000           #6-70
LN.DUM  = 000005           #6-70
LN.LOA  = 000004           #6-70
LN.LOO  = 000003           #6-70
LN.OAU  = 000003           #6-70
LN.OFF  = 000001           #6-70
LN.ON   = 000000           #6-70
LN.OOP  = 000004           #6-70
LN.OPE  = 000001           #6-70
LN.REF  = 000002           #6-70
LN.SER  = 000002           #6-70
LN.STA  = 000017           #6-70
LN.SUB  = 000360           #6-70
LN.TRI  = 000006           #6-70
LP$MPT  = 000010           #7-81
LP$MUX  = 000004           #7-81
LP$TRB  = 000002           #7-81
LP$UNT  = 000001           #7-81
LP$WCN  = 000040           #7-81        7-81
LP$WDV  = 000020           #7-81        7-81
LP$WLD  = 000360           #7-81
LP$WTR  = 000200           #7-81        7-81
LP$WUN  = 000100           #7-81        7-81
LTIM      000014 R         #8-95       *9-153       9-216       9-221       9-222      *14-593
L$ADJ     000110           #6-81       *14-548     *14-564      15-628     *15-629
L$BUF     000060           #6-81        14-575      14-580
L$CHN     000054           #6-81        10-360      15-634
L$COU     000114           #6-81       *14-547     *14-561     *14-562     *14-563      15-626     *15-630
L$CTB     000053           #6-81
L$CTL     000002           #6-81
L$DDM     000000           #6-81        9-229       9-262       9-284       9-303       20-900
L$FLG     000064           #6-81       *9-144       9-149       9-159       9-198       9-223       9-241      *9-329       9-330
                           *9-332       14-525      14-558     *14-574     *14-579
L$FLX     000070           #6-81
L$LTM     000070           #6-81        14-552      14-570     *15-640
L$MSG     000056           #6-81
L$MTYP    000067           #6-81
L$NAM     000000           #6-81
L$NLEN  = 000020           #6-81        6-81        6-81
L$NOD     000112           #6-81        14-553      14-571     *15-637
L$NXT     000044           #6-81
L$OPT     000062           #6-81
L$PAR     000120           #6-81
L$PDV     000055           #6-81        10-353      14-517
L$PFG     000041           #6-81
L$PLB     000116           #6-81
L$PRO     000067           #6-81
L$PVC     000004           #6-81
L$SCN     000020           #6-81
L$SCR     000072           #6-81        16-687
```

```
                                                                        L 13
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16        PAGE 4
```

```
        000002                          B$7:     .BLKB   1                           ; BASE TABLE + 7
        000003                          B$10:    .BLKB   1                           ; BASE TABLE + 10
        000004                          B$11:    .BLKB   1                           ; BASE TABLE + 11
        000005                          B$12:    .BLKB   1                           ; BASE TABLE + 12
        000006                          B$7S10:  .BLKB   1                           ; SUM OF BASE TABLE 7 AND 10
                                                 .EVEN
        000000                                   .PSECT
```

```
                                                                          K 15
CIRECO     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17      PAGE 4
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

ZF.DDM  = 000001           #5-64
ZF.DIA  = 004000           #5-64
ZF.DLC  = 000002           #5-64
ZF.DVP  = 100000           #5-64
ZF.INI  = 040000           #5-64
ZF.KMX  = 000020           #5-64
ZF.LLC  = 000004           #5-64
ZF.LMC  = 000100           #5-64
ZF.MAN  = 020000           #5-64        10-164
ZF.MFL  = 000010           #5-64
ZF.MTM  = 000400           #5-64
ZF.MUX  = 000040           #5-64
ZF.PSE  = 002000           #5-64
ZF.SLI  = 010000           #5-64
ZF.TIM  = 000200           #5-64
ZF.X3P  = 000000           #5-64
ZS.ASN  = 100000           #5-64
ZS.BSY  = 140000           #5-64
Z.AVL     000014           #5-64
Z.DAT     000016           #5-64
Z.DSP     000000           #5-64        5-64
Z.FLG     000010           #5-64        10-154      10-164
Z.LEN   = 000016           #5-64
Z.LLN     000006           #5-64
Z.MAP     000020           #5-64
Z.NAM     000004           #5-64        10-157
Z.PCB     000012           #5-64
Z.SCH     000007           #5-64
$PRCCO  = ******  GX        10-166
```

L 15

```
CIREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17     PAGE 1
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE             REFERENCES

BIT0     = 000001          #5-62
BIT1     = 000002          #5-62
BIT10    = 002000          #5-62
BIT11    = 004000          #5-62
BIT12    = 010000          #5-62
BIT13    = 020000          #5-62
BIT14    = 040000          #5-62
BIT15    = 100000          #5-62
BIT2     = 000004          #5-62
BIT3     = 000010          #5-62
BIT4     = 000020          #5-62
BIT5     = 000040          #5-62
BIT6     = 000100          #5-62
BIT7     = 000200          #5-62
BIT8     = 000400          #5-62
BIT9     = 001000          #5-62
B$10       000003          #5-62
B$11       000004          #5-62
B$12       000005          #5-62
B$5        000000          #5-62
B$6        000001          #5-62
B$7        000002          #5-62
B$7S10     000006          #5-62
CHKCIR   = ******    GX     6-134
CIREIN     000000 RG       #6-93
HEADR    = ******    GX     6-153
H.LUN      000076           6-154
LF$END   = 000020          #5-62       6-160
LF$MLT   = 040000          #5-62
LF$REA   = 000001          #5-62       6-162
LF$SEG   = 100000          #5-62
LF$SIG   = 000040          #5-62       6-131
LF$SKP   = 000004          #5-62
LF$VR2   = 000010          #5-62
LF$ZER   = 000002          #5-62
LP$MPT   = 000010          #5-62
LP$MUX   = 000004          #5-62
LP$TRB   = 000002          #5-62
LP$UNT   = 000001          #5-62
LP$WCN   = 000040          #5-62       5-62
LP$WDV   = 000020          #5-62       5-62
LP$WLD   = 000360          #5-62
LP$WTR   = 000200          #5-62       5-62
LP$WUN   = 000100          #5-62       5-62
L$ADJ      000110          #5-62
L$BUF      000060          #5-62      *6-96        6-161
L$CHN      000054          #5-62       6-107
L$COU      000114          #5-62
L$CTB      000053          #5-62
L$CTL      000002          #5-62
L$DDM      000000          #5-62
L$FLG      000064          #5-62      *6-131      *6-160      *6-162      *6-169
```

```
                                                                  L  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 6
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL  VALUE              REFERENCES

L$PRO      000075          #6-72
L$PVC      000004          #6-72
L$RET      000256          #6-72          *26-1011    *26-1012
L$SCN      000020          #6-72
L$SCR      000166          #6-72
L$SER      000230          #6-72
L$SLEN     000074          #6-72
L$SLT      000042          #6-72
L$SNM      000046          #6-72
L$STA      000100          #6-72
L$TAD      000103          #6-72          *14-727
L$TH1      000130          #6-72          *38-1251
L$TH2      000131          #6-72          *39-1269
L$TH3      000132          #6-72          *40-1287
L$TNM      000052          #6-72
L$TPT      000050          #6-72
L$TRB      000040          #6-72
L$TYP      000063          #6-72           10-549
L$UNT      000003          #6-72
L$XCH      000232          #6-72          *20-886     *20-887      20-888
L$XMT      000120          #6-72          *31-1125
L.COST     000015          #5-67
L.CTL      000012          #5-67
L.CVA      177776          #5-67
L.DDM      000002          #5-67
L.DDS      000004          #5-67
L.DLC      000003          #5-67
L.DLM      000006          #5-67
L.DLS      000010          #5-67
L.FLG      000000          #5-67
L.KRBA     000016          #5-67
L.LEN   =  000022          #5-67
L.MPF      000022          #5-67
L.NMST     000020          #5-67
L.NSTA     000014          #5-67
L.OWNR     000021          #5-67
L.UNT      000013          #5-67
MAXCHN  =  007777          #7-76           20-888
ME$CIR     000003            10-545        15-758      15-763
ME.CST     177765            15-762
ME.FOR     177776            10-555
ME.FUN     177777            10-538
ME.IID     177767            10-548        15-757
ME.PNA     177752            15-750
ME.PVA     177760            15-753
ME$ACT     177776            10-549
MO$CLE     000100            10-539
MS.MOR     000002            10-567
MXBACT     001400 R        #41-1304
MXDTE   =  000017          #7-77           22-931      23-957
MXRACT     001042 R        #24-975
MXWACT     001054 R        #25-992
```

```
                                                                  M  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11      PAGE 7
```

```
          000020                          LP$WDV = 20              ; WILD CARD DEVICE NAME FOUND
          000040                          LP$WCN = 40              ; WILD CARD CONTROLLER NUMBER FOUND
          000100                          LP$WUN = 100             ; WILD CARD UNIT NUMBER FOUND
          000200                          LP$WTR = 200             ; WILD CARD TRIBUTARY NUMBER FOUND
          000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK

                                  ;
                                  ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                  ;

          000001                          LS$NTL=1                 ; NTL SET FUNCTION
          000002                          LS$STA=2                 ; SET STATE
          000004                          LS$COS=4                 ; SET COST
          000010                          LS$OWN=10                ; SET OWNER
          000020                          LS$TAD=20                ; SET TRIBUTARY ADDRESS
          000040                          LS$ACT=40                ; SET MULTIPOINT ACTIVE
          000100                          LS$DEA=100               ; SET MULTIPOINT DEAD
          000200                          LC$NTL=200               ; NTL CLEAR FUNCTION
          000400                          LC$OWN=400               ; CLEAR OWNER
          001000                          LS$DDT=1000              ; SET DEAD TIMER
          002000                          LS$DLT=2000              ; SET DELAY TIMER
          004000                          LS$PLT=4000              ; SET POLL TIMER
          010000                          LS$BBT=10000             ; SET BABBLE TIMER
          020000                          LS$NMT=20000             ; SET NORMAL TIMER
          040000                          LS$XMT=40000             ; SET TRANSMIT TIMER
          100000                          LS$SER=100000            ; SET SERVICE [ENABLE/DISABLE]

                                  ;
                                  ; FLAGS WORD BIT DEFINITIONS (L$FL1)
                                  ;

          000001                          LS$BSA=1                 ; SET ACTIVE BASE
          000002                          LS$BSD=2                 ; SET DYING BASE
          000004                          LS$BSI=4                 ; SET INACTIVE BASE
          000010                          LS$INA=10                ; SET ACTIVE INCREMENT
          000020                          LS$IND=20                ; SET DYING INCREMENT
          000040                          LS$INI=40                ; SET INACTIVE INCREMENT
          000100                          LS$TH1=100               ; SET DEAD THRESHOLD
          000200                          LS$TH2=200               ; SET DYING THRESHOLD
          000400                          LS$TH3=400               ; SET INACTIVE THRESHOLD
          001000                          LS$MXB=1000              ; SET MAXIMUM BLOCKS
          010000                          LS$HTM=10000             ; SET HELLO TIMER
          020000                          LS$MDT=20000             ; SET MODEM TEST
          040000                          LS$LOO=40000             ; SET CONTROLLER LOOPBACK
          100000                          LS$NOR=100000            ; SET CONTROLLER NORMAL
                                          .IF DF  R$$PRO  ; PRO/DECNET
                                  ;
                                  ; LOOPBACK TYPE WORD BIT DEFINITIONS (L$LTY)
                                  ;

                                          LS$INT = 0               ; INTERNAL LOOPBACK
                                          LS$EXT = 1               ; EXTERNAL LOOPBACK
                                  ;
                                  ; MODEM TEST TYPE WORD BIT DEFINITIONS (L$MDT)
                                  ;

                                          LS$LCL = 0               ; LOCAL MODEM TEST
                                          LS$REM = 1               ; REMOTE MODEM TEST
                                          LS$OFF = 2               ; SHUT OFF MODEM TESTS
                                          .ENDC   ; DF R$$PRO
                                  ;
                                  ; Flags word bit definitions for X.25 circuits and lines (L$FLX)
                                  ;
```

```
Symbol table

MP$ALB   000036     MP$DLI=  001466     MP$MAR   001635     MP$PCO   000024     MP$SVR   000163
MP$AMC   001640     MP$DLT   002200     MP$MAV   006321     MP$PCT   000144     MP$SWI   002260
MP$AMH   001641     MP$DRO   001441     MP$MBN   001636     MP$PDT   002114     MP$SYP   002424
MP$ANB   004420     MP$DST   000454     MP$MBR   001637     MP$PHA   000170     MP$TFL   000156
MP$ASB   000024     MP$DTE   002140     MP$MBU   001642     MP$PHY   000012     MP$TH1   002206
MP$ASC   004432     MP$DTY=  001452     MP$MCB   000156     MP$PLI   000025     MP$TH2   002205
MP$AUS   000512     MP$DUA   000207     MP$MCO   001632     MP$PLN   002140     MP$TH3   002202
MP$BBT   002165     MP$DUC   000210     MP$MDE   004421     MP$PLO   000012     MP$TLN   000202
MP$BDF   002164     MP$DUM   000202     MP$MHO   001633     MP$PMC   004374     MP$TLO   000172
MP$BFQ   002121     MP$DUP   002127     MP$MLB   000202     MP$PNT   002126     MP$TPA   004375
MP$BLK   001616     MP$DVC   002114     MP$MLK   001306     MP$PRI   004411     MP$TRI   002164
MP$BLO   001452     MP$DWE   001321     MP$MLN   001631     MP$PRO   002130     MP$TST   000144
MP$BMX   002176     MP$ELT   000157     MP$MLP   006333     MP$PSS   001763     MP$TYP   002130
MP$BNP   006330     MP$ETY   001605     MP$MRB   002171     MP$PST   001762     MP$UCS   004407
MP$BRT   001620     MP$EVE   000311     MP$MRO   001605     MP$RET   001631     MP$USR   001750
MP$BSA   002176     MP$FNC   001752     MP$MRP   006322     MP$RFA   001323     MP$VEC   004410
MP$BSD   002203     MP$GDT   002222     MP$MRT   002153     MP$RMX   002201     MP$VER   004406
MP$BSI   002200     MP$GNM   002223     MP$MRV   006320     MP$RPA   004374     MP$WDF   002165
MP$BSP   006331     MP$GRO   002115     MP$MSB   000170     MP$RPR   001606     MP$WMX   002177
MP$BUF   000170     MP$GRP   000541     MP$MVE   000145     MP$RRT   002213     MP$XMT   002166
MP$BUP   006324     MP$GTY   002224     MP$MVI   001634     MP$RST   002212     MP$XPF   000242
MP$BUS   001643     MP$HAD   001757     MP$MVR   001751     MP$RSV   001754     MP$XPL   000244
MP$CAC   001750     MP$HBT   002142     MP$MWN   002154     MP$RSZ   001756     MP$XPT   000240
MP$CAS   001762     MP$HDD   000162     MP$MXB   002172     MP$RTI   001616     MP$XXX   177777
MP$CAT   002210     MP$HTM   001612     MP$MXC   000466     MP$RTM   001611     MP$5ST   004406
MP$CCS   004406     MP$HWA   002210     MP$MXR   001630     MP$RVE   001604     MS$ACT   000001
MP$CHN   002141     MP$IAT   001322     MP$MXW   002143     MP$RVT   000156     MS$ADU   000010
MP$CIR   000144     MP$IDE   000144     MP$NAA   006335     MP$SAC   000514     MS$ALO   000007
MP$CLK   002131     MP$IDP   006327     MP$NAC   001130     MP$SAD   000543     MS$ASE   000006
MP$CLN   002126     MP$IHO   000215     MP$NAP   006326     MP$SCA   000310     MS$ATR   000011
MP$CLT   002211     MP$INA   002177     MP$NCT   000240     MP$SCO   000144     MS$AUT   000000
MP$CMB   002142     MP$IND   002204     MP$NET   002114     MP$SCT   002176     MS$CLE   000003
MP$CMK   000537     MP$INI   002201     MP$NLI   000765     MP$SDU   000203     MS$DED   000004
MP$CMX   002153     MP$ITI   000776     MP$NNA   000764     MP$SDV   000160     MS$DIE   000003
MP$CND   000310     MP$LAA   000231     MP$NOD   000500     MP$SEH   004401     MS$DUM   000004
MP$CNU   001753     MP$LAN   000234     MP$NRA   006334     MP$SER   000144     MS$FAI   000013
MP$COB   000311     MP$LAR   000202     MP$NRB   000214     MP$SET   000000     MS$GLO   000000
MP$CON   002126     MP$LAS   000012     MP$NSA   001617     MP$SGZ   001644     MS$HOL   000002
MP$COS   001604     MP$LBS   000203     MP$NTI   002141     MP$SID   000176     MS$INA   000002
MP$CPF   000230     MP$LCO   000226     MP$NUM   001642     MP$SIN   000310     MS$LOA   000003
MP$CPL   000232     MP$LCT   000156     MP$NVE   001274     MP$SLI   000156     MS$LOO   000002
MP$CPT   000226     MP$LHL   000232     MP$NXN=  001476     MP$SLO   000171     MS$OFF   000001
MP$CPU   0[illegible]161     MP$LLE   000227     MP$OAC   000632     MP$SMX   002202     MS$ON    000000
MP$CSZ   001755     MP$LLO   004432     MP$OAN   000620     MP$SND   000500     MS$PRO   000006
MP$CUS   002127     MP$LMB   002152     MP$OCO   000776     MP$SNP   006332     MS$REA   000004
MP$CVA   000540     MP$LMX   002200     MP$OHO   000214     MP$SNU   000542     MS$REF   000001
MP$DAL   002570     MP$LNA   000144     MP$ONA   000764     MP$SOB   000524     MS$RES   000003
MP$DCO=  001464     MP$LNO   000233     MP$ONR   002114     MP$SPA   000157     MS$RST   000001
MP$DDT   002177     MP$LOA   000170     MP$OQL   001453     MP$SPR   000536     MS$RSX   000002
MP$DEL   001131     MP$LOG   004514     MP$OTI   000777     MP$SPS   000513     MS$RT    000005
MP$DES   000156     MP$LOO   000620     MP$OTY   001022     MP$STA   000000     MS$SER   000002
MP$DEV   002114     MP$LTM   001613     MP$OUS   000777     MP$STI   002140     MS$SHU   000002
MP$DFA   001320     MP$LTY   004533     MP$OVE   001010     MP$STT   002201     MS$SRV   000007
MP$DHO=  001465     MP$LWI   000230     MP$OWN   004374     MP$STY   000175     MS$STA   000000
MP$DIA   000173     MP$MAC   004420     MP$PAR   000036     MP$SUB   000001     MS$SYN   000012
MP$DLB   002571     MP$MAD   001630     MP$PAS   000513     MP$SUR   000156     MS$TER   000001
MP$DLG   004521     MP$MAP   006323     MP$PCH   002152     MP$SUS   000512     MS$TOP   000003
```

Table of contents

```
     5-   49     MACRO CALLS AND LOCAL DATA
     6-   84     Context area definition
     7-   88     ERROR MESSAGE STRINGS
     8-  112     SET DMP PARAMETERS TABLES
     9-  147     SETNTL - DISPATCH NETLOADER SET FUNCTIONS
    10-  181     CLENTL - DISPATCH NETLOADER CLEAR FUNCTIONS
    11-  208     NTLLIN - PERFORM NTL SET/CLEAR LINE FUNCTIONS
    12-  267     SETHTM - SET HELLO TIMER
    13-  316     SETSTA - SET A NEW CIRCUIT STATE
    14-  505     STAON - REQUEST CIRCUIT STATE "ON"
    15-  576     STASER - REQUEST CIRCUIT STATE "SERVICE"
    16-  646     STAOFF - REQUEST CIRCUIT STATE "OFF"
    17-  673     TWKLIN - TWEAK A CIRCUIT'S STATE
    18-  708     LINON - TURN CIRCUIT ON
    19-  771     LINOFF - TURN CIRCUIT OFF
    20-  833     SETCOS - SET COST FOR A CIRCUIT
    21-  931     SETOWN - SET CIRCUIT OWNER
    22- 1010     DMP SET ROUTINES
    23- 1058     DMPSET - DMP SET PARAMETERS ROUTINE
    24- 1109     SETSER - Set SERVICE [ENABLE/DISABLE]
```

```
                                                .SBTTL  SETHTM - SET HELLO TIMER

                                        ;+
                                        ;
                                        ; SETHTM - SET HELLO TIMER
                                        ;
                                        ; INPUTS:
                                        ;       L$SLT(R4) - SLT ADDRESS
                                        ;       L$HTM(R4) - HELLO TIMER
                                        ;
                                        ; OUTPUTS:
                                        ;       R0 - SUCCESS CODE
                                        ;
                                        ; N.B. *** THIS CODE CHANGES APR6 MAPPING AND MUST RESIDE IN APR5 SPACE ***
                                        ;
                                        ;-
000566                                  SETHTM::
000566  012700  177752                          MOV     #ME.PNA,R0              ; ASSUME PARAMETER NOT APPLICABLE
000572  012764  001612  000072                  MOV     #MP$HTM,L$PAR(R4)       ; ...
000600  022764  177777  000050                  CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
000606  001450                                  BEQ     40$                     ; BR IF YES
000610  016401  000042                          MOV     L$SLT(R4),R1            ; GET SLT ADDRESS
000614  012700  177765                          MOV     #ME.CST,R0              ; ASSUME LINE IS IN WRONG STATE
000620  012764  000001  000072                  MOV     #ME$LIN,L$PAR(R4)       ; ...
000626  032761  040000  000000                  BIT     #LF.RDY,L.FLG(R1)       ; IS THE LINE LOADED?
000634  001435                                  BEQ     40$                     ; BR IF NO
000636  012700  177752                          MOV     #ME.PNA,R0              ; ELSE ASSUME PARAMETER NOT APPLICABLE
000642  012764  001612  000072                  MOV     #MP$HTM,L$PAR(R4)       ; ...
000650  000241                                  CLC                             ; ASSUME SUCCESS

000652                                          SWSTK$  30$                     ;; ENTER SYSTEM STATE
000656                                          SAVMAP                          ;; SAVE CURRENT MAPPING
000662                                          CALL    FNDCTR                  ;; FIND COUNTER BLOCK
000666  103006                                  BCC     10$                     ;; BR IF FOUND IT
000670                                          RETC    R0                      ;; ELSE SET USER C-BIT
000702  000403                                  BR      20$                     ;; AND EXIT
000704  016461  000226  000030          10$:    MOV     L$HTM(R4),T$T5(R1)      ;; STORE HELLO TIMER
000712                                  20$:    RESMAP                          ;; RESTORE PREVIOUS MAPPING
000716                                          RETURN

000720  103404                          30$:    BCS     50$                     ; EXIT IF ERROR
000722  005064  000072                          CLR     L$PAR(R4)               ; CLEAR ERROR DETAIL
000726  000401                                  BR      50$                     ; AND EXIT
000730  000261                          40$:    SEC                             ; ELSE INDICATE ERROR
000732                                  50$:    RETURN

                                                .ENDC   ; NDF R$$11S

                                        .SBTTL  LINOFF - TURN CIRCUIT OFF
                                ;+
                                ; **-LINOFF-TURN CIRCUIT OFF
                                ;
                                ; THIS ROUTINE IS CALLED TO TURN A CIRCUIT OFF
                                ;
                                ; INPUTS:
                                ;       R4 = ADDRESS OF THE CONTEXT BLOCK
                                ;
                                ; OUTPUTS:
                                ;       IF CC, CIRCUIT TURNED OFF,
                                ;       ELSE, ERROR RETURN IN R0
                                ;
                                ; REGISTERS:
                                ;       R4, R5 ARE PRESERVED
                                ;-
                                        .ENABL  LSB

002414                          LINOFF:

002414                          10$:    SWSTK$  20$                     ;; ENTER SYSTEM STATE
002420  010405                          MOV     R4,R5                   ;; SAVE ADDRESS OF CONTEXT BLOCK
002422                                  CALL    @CCBGT                  ;; TRY TO ALLOCATE A CCB        ; RJK08
002426  103417                          BCS     15$                     ;; IF CS, ALLOCATION FAILURE
002430  116564  000055  000007          MOVB    L$PDV(R5),C.STA(R4)     ;; SET UP PDV AND
002436  116564  000054  000006          MOVB    L$CHN(R5),C.LIN(R4)     ;;   CHANNEL # TO TURN OFF
002444  005064  000004                  CLR     C.NSP(R4)               ;; SET RETURN TO AUX
002450  005064  000012                  CLR     C.STS(R4)               ;; TURN CIRCUIT OFF
002454  012764  002006  000010          MOV     #FC.CTL!FS.STP,C.FNC(R4);; SET UP FUNCTION CODE
002462                                  CALLR   @LLCRS                  ;; QUEUE CCB TO THE LLC         ; RJK08

002466                          15$:    RETC    R0                      ;; SET USER C-BIT
002500                                  RETURN                          ;; LEAVE SYSTEM STATE

002502  103004                  20$:    BCC     25$                     ; IF CC, CCB WAS QUEUED
002504                                  WSIG$S                          ; ALLOCATION FAILURE ... WAIT A BIT
002512  000740                          BR      10$                     ;   THEN TRY AGAIN

002514  016401  000042          25$:    MOV     L$SLT(R4),R1            ; GET SLT ADDRESS
002520  032761  000020  000000          BIT     #LF.MTP,L.FLG(R1)       ; MULTI-POINT CIRCUIT?
002526  001421                          BEQ     45$                     ; BR IF NO
002530  016401  000050                  MOV     L$TPT(R4),R1            ; POINT TO STATION TABLE
002534  142761  000200  000000          BICB    #SF.ACT,S.FLG(R1)       ; TURN THE CIRCUIT OFF
002542  116100  000014          40$:    MOVB    L.NSTA(R1),R0           ; CHECK IF CIRCUIT NO LONGER ACTIVE
002546  010102                          MOV     R1,R2                   ; COPY SLT POINTER
002550  062702  000022                  ADD     #L.MPF,R2               ; POINT TO MULTI-DROP FLAGS BYTE
                                ;
                                ; SCAN ALL STATIONS ON THIS CIRCUIT
                                ;
002554  132712  000200          43$:    BITB    #SF.ACT,(R2)            ; IS THIS STATION ACTIVE ?
002560  001007                          BNE     50$                     ; IF MI, YES - CIRCUIT STILL ACTIVE
002562  062702  000004                  ADD     #S.LEN,R2               ; ELSE, POINT TO NEXT FLAGS BYTE
002566  005300                          DEC     R0                      ; LOOP STATION # TIMES
002570  003371                          BGT     43$                     ;    ...
002572  042761  100000  000000  45$:    BIC     #LF.ACT,L.FLG(R1)       ; CIRCUIT NO LONGER ACTIVE
002600  000241                  50$:    CLC
002602                          60$:    RETURN
```

```
                                                                L 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-5
Symbol table

MU$PER  000000            NM.VR3  002402            N$$MLL= 000001            P$LEN   000052            P.XAQC  000026
MV$III  000000            NODMSG  000074R           N$$MOV= 000010            P$LST   000000            P.XAWQ  000030
MV$IV   000001            NTLLIN  000446R           N$$NCT= 000001            P$NRNI  000040            P.XBFC  000027
MXBTBL  000310R           NTLMSG  000035R           N$$PEM= 000001            P$OCCB  000050            Q$$OPT= 000010
MX$ACT  000020            NTLSND= ****** GX         PDVID = ****** GX         P$PFQ   000006            RF.CTL= 000003
MX$CIR  000020            N$ACQ   000000            PDVTA = ****** GX         P$PKSZ  000044            RF.LD1= 000040
MX$CLN  000020            N$ACTL  000032            PF$BLK= 020000            P$RMX1  000016            RF.LD2= 000100
MX$CNM  000006            N$ADJ1  000072            PF$CLC= 010000            P$RMX2  000020            RF.TIM= 177400
MX$CON  000006            N$ADJ2  000074            PF$DLM= 100000            P$RPRI  000042            RF.TMO= 000400
MX$DAC  000020            N$CACH  000062            PF$EIP= 000002            P$RTIM  000003            RF.WFC= 000200
MX$DTE  000020            N$CIR   000034            PF$ENB= 000001            P$STA1  000022            RF.WTD= 000020
MX$FIL  000034            N$CRC   000120            PF$FAI= 004000            P$STA2  000023            RF.WTM= 000030
MX$LGC  000777            N$DLA   000020            PF$FM1= 000100            P$TIM   000010            RF.WTS= 000010
MX$LIN  000020            N$DLY   000014            PF$FM2= 000200            P$TSCT  000026            R$$DER= 000000
MX$LON  000006            N$ELEN  000054            PF$OFF= 000000            P$TSIZ  000024            R$$K11= 000001
MX$NMS  000454            N$ENC   000042            PF$RM1= 000020            P$TYP   000001            R$$SND= 000000
MX$NOD  000006            N$ERRC  000022            PF$RM2= 000040            P$$P45= 000000            R$$11M= 000000
MX$OBJ  000006            N$FLG   000005            PF$RVR= 000010            P$$WRD= 000000            SETBBT  000222RG      002
MX$OWN  000040            N$FNC   000006            PF$STA= 000004            P.ACTC  000012            SETBSA  000260RG      002
MX$PAR  000036            N$GENQ  000052            PF$SVC= 040000            P.CFNC  000034            SETBSD  000272RG      002
MX$PAS  000010            N$GTM   000015            PF$UP = 000004            P.CHAR  000042            SETBSI  000304RG      002
MX$RAC  000047            N$HC1   000052            PR$BED= 000200            P.CONT  000067            SETCOS  002604RG
MX$RID  000047            N$HC2   000056            PR$BEU= 000100            P.DEDT  000100            SETHTM  000566RG
MX$RPS  000047            N$HIGH  000033            PR$BRD= 000040            P.DELT  000076            SETINA  000316RG      002
MX$SID  000040            N$LLT   000026            PR$BRU= 000020            P.DISL  000006            SETIND  000330RG      002
MX$SNK  000377            N$LLTM  000024            PR$DWN= 000002            P.DISR  000004            SETINI  000342RG      002
MX$TYP  000077            N$LVC   000036            PR$LCC= 000010            P.DVRF  000047            SETMXB  000412RG      002
MX$UID  000020            N$LV1   000002            PR$MOP= 000004            P.DVTM  000046            SETNMT  000234RG      002
M$$CRB= 000124            N$LV2   000010            PR$UP = 000001            P.ECCB  000064            SETNTL  000322RG
M$$CRX= 000000            N$MBXQ  000050            PS$CHR= 000016            P.ENBC  000011            SETOWN  000000RG      002
M$$FCS= 000000            N$MHC1  000036            PS$FAI= 000014            P.HDER  000056            SETSER  000566RG      002
M$$MGE= 000000            N$MHC2  000044            PS$NTI= 000006            P.INPQ  000036            SETSTA  000734RG
M$$NET= 000000            N$PLD   000016            PS$OFF= 000000            P.LEN = 000154            SETTH1  000354RG      002
M$$OVR= 000000            N$PLLT  000030            PS$STR= 000002            P.LERR  000054            SETTH2  000366RG      002
NF$ACC= 100000            N$PRI   000076            PS$UP = 000012            P.LINE  000010            SETTH3  000400RG      002
NF$BLK= 000100            N$ROA1  000022            PS$VER= 000010            P.MODE  000060            SETXMT  000246RG      002
NF$DMO= 000010            N$ROA2  000030            PS$WT = 000004            P.MODI  000061            SF$ANM  000001
NF$EVT= 040000            N$RTMX  000014            PT$BRO= 000200            P.MTER  000057            SF$DNM  000002
NF$MOU= 000040            N$RTM1  000014            PT$DRT= 000100            P.OFFS= 000046 G          SF.ACT= 000200
NF$RST= 000002            N$RTM2  000015            PT$END= 000004            P.PARM  000070            SF.ENA= 000100
NF$SCN= 000020            N$RT1   000000            PT$LV1= 000002            P.PIND  000066            SF.LPB= 000004
NF$SHU- 000004            N$RT2   000006            PT$LV2= 000001            P.POLT  000102            SF.MFL= 000040
NF$TIM= 000200            N$SLA   000016            PT$PH3= 000010            P.RAQC  000020            SF.PAC= 000020
NF.ACC= 000001            N$SNOD  000012            PT$XAR= 000020            P.RASQ  000022            SF.REA= 000010
NF.BCP= 000040            N$TCTL  000112            PX$BLK= 000040            P.RBFC  000021            SF.SER= 000001
NF.END= 000200            N$TIM   000004            PX$DLM= 000200            P.RERR  000052            SF.SVC= 000002
NF.EVT= 000002            N$TLC   000100            PX$SVC= 000100            P.SEL   177776            SF.UNL= 000040
NF.LLI= 000004            N$TNC   000114            P$CHN   000004            P.SPFC  000104            SLTMA = ****** GX
NF.LV2= 000100            N$TRC   000106            P$CNT   000005            P.STAT  000002            STAOFF  002100R
NF.SLI= 000400            N$TTCB  000110            P$CTR   000034            P.STMT  000074            STAON   001534R
NF.SMC= 000020            N$VCB   000010            P$DRCT  000015            P.TBAD  000062            STASER  001706R
NF.SWP= 000010            N$VER   000066            P$DRTR  000036            P.TIMC  000050            S$$WRG= 000000
NMTTBL  000132R           N$XLEN  000124            P$FLG   000012            P.TIME  000001            S$$YSZ= 007600
NM.CLN  003400            N$$ACC= 000001            P$FWD   000030            P.TIMR  000000            S.COST  000001
NM.INI  002400            N$$BUF= 000001            P$ICCB  000046            P.TRIB  000014            S.FLG   000000
NM.OPR  003000            N$$LDV= 000001            P$IPL   000014            P.VECT  177732            S.LEN   000004
NM.VR2  002401            N$$MCP= 000001            P$LCD   000002            P.VEC1  177752            S.NMST  000002
```

```
                                                                M 7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-6
```

```
**FILE**ID**CICHPX




  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH   PPPPPPPP     XX      XX
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH   PPPPPPPP     XX      XX
CC               II       CC           HH      HH   PP      PP   XX      XX
CC               II       CC           HH      HH   PP      PP   XX      XX
CC               II       CC           HH      HH   PP      PP     XX  XX
CC               II       CC           HH      HH   PP      PP     XX  XX
CC               II       CC           HHHHHHHHHH   PPPPPPPP         XX
CC               II       CC           HHHHHHHHHH   PPPPPPPP         XX
CC               II       CC           HH      HH   PP             XX  XX
CC               II       CC           HH      HH   PP             XX  XX
CC               II       CC           HH      HH   PP           XX      XX     ....
CC               II       CC           HH      HH   PP           XX      XX     ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH   PP           XX      XX     ....
  CCCCCCCC     IIIIII       CCCCCCCC   HH      HH   PP           XX      XX     ....




LL             SSSSSSSS   TTTTTTTTTT
LL             SSSSSSSS   TTTTTTTTTT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL           SS               TT
LL             SSSSSS         TT
LL             SSSSSS         TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LL                   SS       TT
LLLLLLLLLL   SSSSSSSS         TT
LLLLLLLLLL   SSSSSSSS         TT
```

```
                                               .SBTTL  SETCTR - Set flags in Line Counter block
                                       ;+
                                       ; *** SETCTR - Set flags in Line Counter block
                                       ;
                                       ; INPUTS
                                       ;       R1 - Address of line counters block
                                       ;       R4 - Address of context area
                                       ;
                                       ; OUTPUTS
                                       ;       The Blocking Flag in the line counters block is set as required
                                       ;-

 000662                                SETCTR:
 000662  032764  001000  000076                BIT     #LS$BLK,L$FLX(R4)       ;; Is the blocking parameter set?
 000670  001412                                BEQ     10$                     ;; Branch if not to be changed
 000672  142761  000040  000044                BICB    #PX$BLK,T$FLAG(R1)      ;; Assume blocking disabled
 000700  022764  000000  000262                CMP     #MB$ENA,L$BLK(R4)       ;; Is blocking to be enabled?
 000706  001003                                BNE     10$                     ;; If not, we have done right
 000710  152761  000040  000044                BISB    #PX$BLK,T$FLAG(R1)      ;; Else, restore blocking flag
 000716                                10$:    RETURN
```

```
                                                                      L 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15      PAGE 4
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE            REFERENCES

L$MWN     000252          #6-104
L$MXB     000133          #6-104
L$MXR     000253          #6-104      7-126      7-127
L$MXW     000252          #6-104      7-124
L$NAM     000000          #6-104
L$NLEN  = 000020          #6-104      6-104      6-104
L$NMT     000116          #6-104
L$NTI     000256          #6-104
L$NTL     000134          #6-104
L$NUM     000240          #6-104      7-130
L$NUML    000236          #6-104      7-125
L$NXT     000044          #6-104
L$OPT     000062          #6-104
L$OWN     000102          #6-104
L$PAR     000072          #6-104     *8-165     *8-169     *8-173     *9-199     *10-235    *10-242    *11-294    *15-407
L$PDV     000055          #6-104
L$PFG     000041          #6-104      11-265     11-278
L$PLT     000112          #6-104
L$PRO     000075          #6-104
L$PVC     000004          #6-104
L$RET     000256          #6-104      7-128      7-129
L$SCN     000020          #6-104
L$SCR     000166          #6-104
L$SER     000230          #6-104
L$SLEN    000074          #6-104
L$SLT     000042          #6-104      11-261     15-392
L$SNM     000046          #6-104
L$STA     000100          #6-104
L$TAD     000103          #6-104
L$TH1     000130          #6-104
L$TH2     000131          #6-104
L$TH3     000132          #6-104
L$TNM     000052          #6-104
L$TPT     000050          #6-104      8-148      11-280     15-386
L$TRB     000040          #6-104
L$TYP     000063          #6-104
L$UNT     000003          #6-104
L$XCH     000232          #6-104
L$XMT     000120          #6-104
L.COST    000015          #5-72
L.CTL     000012          #5-72
L.CVA     177776          #5-72
L.DDM     000002          #5-72
L.DDS     000004          #5-72
L.DLC     000003          #5-72
L.DLM     000006          #5-72
L.DLS     000010          #5-72
L.FLG     000000          #5-72       11-262     11-270
L.KRBA    000016          #5-72
L.LEN   = 000022          #5-72
L.MPF     000022          #5-72
L.NMST    000020          #5-72       11-272
```

```
                                                                      M 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15      PAGE 5
```

```
000406  001436                          BEQ     45$                     ; if EQ, none
000410  004567  000000G                 JSR     R5,SETPAR               ; LISTEN TIMER PARAMETER
000414  001613                          .WORD   MP$LTM                  ; ...
000416  000002                          .WORD   2
000420  116723  177370                  MOVB    LTIM,(R3)+              ; STORE LOW BYTE OF LISTEN TIMER
000424  116723  177365                  MOVB    LTIM+1,(R3)+            ; STORE HIGH BYTE
000430  032764  040000  000064          BIT     #LF$MLT,L$FLG(R4)       ; multiple adjacency sequence ?
000436  001402                          BEQ     40$                     ; if EQ, no
000440  000167  000372                  JMP     80$                     ; if NE, yes - skip other parameters

                                ;
                                ; RETURN BLOCKING, MAXIMUM RECALLS, RECALL TIMER
                                ;
000444  026427  000000  015355  40$:    CMP     L$DDM(R4),#^RDLM        ; IS THIS A DLM CIRCUIT?
000452  001014                          BNE     45$                     ; BR IF NO
000454  004567  000000G                 JSR     R5,SETPAR               ; BLOCKING PARAMETER
000460  001616                          .WORD   MP$BLK                  ; ...
000462  000201                          .WORD   MT$COD!1                ; ONE BYTE CODED
000464  116723  177326                  MOVB    BLKING,(R3)+            ; Store blocking value
000470  032767  000001  177322          BIT     #FL.SVC,FLAG            ; IS THIS AN SVC ?
000476  001402                          BEQ     45$                     ; BR IF NO
000500                                  CALL    SVCRCL                  ; GET SVC RECALL VALUES
                                ;
                                ; RETURN CIRCUIT OWNER
                                ;
000504  032764  040000  000064  45$:    BIT     #LF$MLT,L$FLG(R4)       ; multiple adjacency sequence ?
000512  001151                          BNE     80$                     ; if NE, yes - skip remaining params
000514  116002  000021                  MOVB    L.NMST+1(R0),R2         ; ASSUME OWNER IS IN LINE SLT
000520  105760  000014                  TSTB    L.NSTA(R0)              ; MULTIPOINT ?
000524  001402                          BEQ     50$                     ; IF EQ, NO
000526  116102  000003                  MOVB    S.NMST+1(R1),R2         ; ELSE, YES - GET OWNER FROM TRIB EXTENTIONS
000532  005702                  50$:    TST     R2                      ; IS THERE AN OWNER SET ?
000534  001450                          BEQ     60$                     ; IF EQ, NO OWNER - SKIP IT
000536  004567  000000G                 JSR     R5,SETPAR               ; SET OWNER PARAMETER
000542  002114                          .WORD   MP$ONR                  ; ...
000544  000100                          .WORD   MT$ASC                  ; ASCII IMAGE FIELD
000546  067702  000000G                 ADD     @PDVTA,R2               ; INDEX INTO THE PDV TABLE      ; RJK10
000552  011202                          MOV     (R2),R2                 ; GET THE PDV ADDRESS
000554                                  SAVRG   <R0,R1>                 ; SAVE SOME REGISTERS
000560  016201  000004                  MOV     Z.NAM(R2),R1            ; GET THE PROCESS NAME
000564  010167  177212                  MOV     R1,OWNER                ; save it
000570                                  CALL    FMTOWN                  ; FORMAT THE IMAGE FIELD
000574                                  RESRG   <R1,R0>                 ; RESTORE REGISTERS
                                ;
                                ; RETURN USAGE
                                ;
000600  026427  000000  015355          CMP     L$DDM(R4),#^RDLM        ; IS THIS DLM?
000606  001023                          BNE     60$                     ; BR IF NO
000610  112702  000000                  MOVB    #MU$PER,R2              ; ASSUME PVC
000614  032767  000001  177176          BIT     #FL.SVC,FLAG            ; IS THIS A DLM SVC?
000622  001410                          BEQ     55$                     ; BR IF NO
000624  112702  000001                  MOVB    #MU$INC,R2              ; ASSUME INCOMING SVC
000630  032767  000004  177162          BIT     #FL.INC,FLAG            ; INCOMING SVC?
000636  001002                          BNE     55$                     ; BR IF YES
000640  112702  000002                  MOVB    #MU$OUT,R2              ; ELSE MUST BE OUTGOING SVC
000644  004567  000000G         55$:    JSR     R5,SETPAR               ; SET USAGE PARAMETER
000650  002127                          .WORD   MP$CUS                  ; ...
000652  000201                          .WORD   MT$COD!1                ; ONE BYTE CODED FIELD
```

```
                                                .SBTTL  SVCRCL - Format SVC recall values

                                        ;+
                                        ;
                                        ; SVCRCL - MAXIMUM RECALLS, RECALL TIMER
                                        ;
                                        ; INPUTS:
                                        ;       R3 - POINTER TO NEXT AVAILABLE BYTE IN OUTPUT BUFFER
                                        ;       R4 - ADDRESS OF CONTEXT AREA
                                        ;       SVCMAP - SVC DESCRIPTOR MAPPING BIAS
                                        ;       SVCADD - SVC DESCRIPTOR VIRTUAL ADDRESS
                                        ;
                                        ; OUTPUTS:
                                        ;       FLAG - FL.INC SET IF INCOMING SVC
                                        ;       R3 - UPDATED
                                        ;
                                        ;-
 003006                                 SVCRCL: SWSTK$  20$                             ;; ENTER SYSTEM STATE
 003012                                         SAVMAP                                  ;; SAVE CURRENT MAPPING
 003016                                         MAP     SVCMAP                          ;; MAP TO SVC DESCRIPTOR
 003024 016701  174756                          MOV     SVCADD,R1                       ;; GET ADDRESS OF SVC DESCRIPTOR
 003030 004567  000000G                         JSR     R5,SETPAR                       ;; SET MAXIMUM RECALLS
 003034 001630                                  .WORD   MP$MXR                          ;; ...
 003036 000001                                  .WORD   1                               ;; 1 BYTE DECIMAL FIELD
 003040 116123  000013                          MUVB    S$RCL(R1),(R3)+                 ;; STORE MAXIMUM RECALLS
 003044 004567  000000G                         JSR     R5,SETPAR                       ;; SET RECALL TIMER
 003050 001631                                  .WORD   MP$RET                          ;; ...
 003052 000002                                  .WORD   2                               ;; 2 BYTE DECIMAL FIELD
 003054 116123  000016                          MOVB    S$TMR(R1),(R3)+                 ;; STORE RECALL TIMER
 003060 116123  000017                          MOVB    S$TMR+1(R1),(R3)+               ;; ...
 003064 032761  000002  000010                  BIT     #SF$INC,S$FLG(R1)               ;; IS THIS AN INCOMING SVC?
 003072 001403                                  BEQ     10$                             ;; BR IF NO
 003074 052767  000004  174716                  BIS     #FL.INC,FLAG                    ;; ELSE INDICATE INCOMING SVC
 003102                                 10$:    RESMAP                                  ;; RESTORE PREVIOUS MAPPING
 003106 010366  000010                          MOV     R3,10(SP)                       ;; UPDATE USER R3
 003112                                 20$:    RETURN
```

```
SYMBOL CROSS REFERENCE                                    CREF    04.00

SYMBOL  VALUE            REFERENCES

L$SLEN    000066          #6-81
L$SLT     000042          #6-81        9-123       17-728
L$SNM     000046          #6-81
L$SYL     000122          #6-81
L$TNM     000052          #6-81
L$TPT     000050          #6-81        9-124       17-736
L$TRB     000040          #6-81        17-748
L$TSZ     000070          #6-81
L$TYP     000063          #6-81
L$UNT     000003          #6-81
L$XLEN    000124          #6-81
L.COST    000015          #6-70        9-186
L.CTL     000012          #6-70
L.CVA     177776          #6-70
L.DDM     000002          #6-70
L.DDS     000004          #6-70
L.DLC     000003          #6-70
L.DLM     000006          #6-70
L.DLS     000010          #6-70
L.FLG     000000          #6-70        9-142       9-180       9-287       9-290       9-320       17-730      17-732      20-908
L.KRBA    000016          #6-70
L.LEN   = 000022          #6-70
L.MPF     000022          #6-70
L.NMST    000020          #6-70        9-243
L.NSTA    000014          #6-70        9-187       9-244       9-293       17-734      20-906
L.OWNR    000021          #6-70        17-729
L.UNT     000013          #6-70
MAPADD  = ******   GX      16-689
MB$DIS    000001           13-479      20-915
MB$ENA    000000           13-482      20-905
MP$ADJ    001440           9-158
MP$BLK    001616           9-232
MP$CMB    002142           19-872
MP$COS    001604           9-183
MP$CUS    002127           9-272
MP$DRO    001441           9-178
MP$HTM    001612           9-211
MP$LTM    001613           9-219
MP$MAC    004420           11-415
MP$MRO    001605           9-201
MP$MXR    001630           18-822
MP$MXW    002143           19-880
MP$NUM    001642           19-863
MP$ONR    002114           9-250
MP$RET    001631           18-826
MP$RPR    001606           9-205
MP$SER    000144           20-903
MP$TRI    002164           11-402
MP$TYP    002130           9-281
MS.PAR    000003           14-575
MS.SUC    000001           14-580
MTPCHA    001164 R         9-322       #11-383
```

```
                                        .SBTTL  BIT DEFINITIONS
                                ;
                                ; PARSE FLAG DEFINITIONS (L$PFG)
                                ;
        000001                          LP$UNT = 1                      ; UNIT NUMBER FOUND
        000002                          LP$TRB = 2                      ; TRIBUTARY NUMBER FOUND
        000004                          LP$MUX = 4                      ; DEVICE IS MUX
        000010                          LP$MPT = 10                     ; LINE IS MULTIPOINT
        000020                          LP$WDV = 20                     ; WILD CARD DEVICE NAME FOUND
        000040                          LP$WCN = 40                     ; WILD CARD CONTROLLER NUMBER FOUND
        000100                          LP$WUN = 100                    ; WILD CARD UNIT NUMBER FOUND
        000200                          LP$WTR = 200                    ; WILD CARD TRIBUTARY NUMBER FOUND
        000360                          LP$WLD = LP$WDV!LP$WCN!LP$WUN!LP$WTR ; WILD CARD FIELD MASK
                                ;
                                ; FLAGS WORD BIT DEFINITIONS (L$FLG)
                                ;
        000001                          LF$REA = 1                      ; READ COUNTERS OPERATION
        000002                          LF$ZER = 2                      ; ZERO COUNTERS OPERATION
        000004                          LF$SKP = 4                      ; SKIP NEXT "FIND NEXT LINE" OPERATION.
                                                                        ; THIS IS USED TO FORCE AN EXTRA PASS
                                                                        ; FOR A MULTIPOINT LINE TO RETURN THE
                                                                        ; CONTROLLER COUNTERS AS WELL AS ALL
                                                                        ; OF THE TRIBUTARY COUNTERS.
        000010                          LF$VR2 = 10                     ; CONNECTED TO VERSION 2.0 NCP
        000020                          LF$END = 20                     ; EXECUTOR IS AN ENDNODE
        000040                          LF$SIG = 40                     ; SHOWING SIGNIFIGANT LINES/CIRCUITS
        040000                          LF$MLT = 40000                  ; MULTIPLE ADJACENCY FLAG
        100000                          LF$SEG  = 100000                ; SEGMENTED RESPONSE IN PROGRESS
                                ;
                                ; BIT POSITION DEFINITIONS
                                ;
        000001                          BIT0 = 1
        000002                          BIT1 = 2
        000004                          BIT2 = 4
        000010                          BIT3 = 10
        000020                          BIT4 = 20
        000040                          BIT5 = 40
        000100                          BIT6 = 100
        000200                          BIT7 = 200
        000400                          BIT8 = 400
        001000                          BIT9 = 1000
        002000                          BIT10 = 2000
        004000                          BIT11 = 4000
        010000                          BIT12 = 10000
        020000                          BIT13 = 20000
        040000                          BIT14 = 40000
        100000                          BIT15 = 100000
```

```
CIRECO     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17     PAGE 5
MACRO CROSS REFERENCE                                   CREF   04.00

MACRO NAME       REFERENCES

BIAS             #5-59
CALL              10-144      10-159
CALLR             10-166
CIRCX$           #5-60        6-69
MANDF$           #5-60        5-62
MAP              #5-59
PDVDF$           #5-60        5-64
RESMAP           #5-59
RESRG            #5-59
RETURN            10-172
SAVMAP           #5-59
SAVRG            #5-59
SLTDF$           #5-60        5-63
```

```
CIREIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:17     PAGE 2
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE            REFERENCES

L$FLX      000070         #5-62
L$LEN      000124         #5-62       6-101
L$LTM      000070         #5-62
L$MSG      000056         #5-62
L$MTYP     000067         #5-62
L$NAM      000000         #5-62
L$NLEN   = 000020         #5-62       5-62        5-62
L$NOD      000112         #5-62
L$NXT      000044         #5-62
L$OPT      000062         #5-62      *6-95        6-114
L$PAR      000120         #5-62      *6-133       6-166       6-167
L$PDV      000055         #5-62
L$PFG      000041         #5-62
L$PLB      000116         #5-62
L$PRO      000067         #5-62
L$PVC      000004         #5-62
L$SCN      000020         #5-62
L$SCR      000072         #5-62
L$SLEN     000066         #5-62
L$SLT      000042         #5-62
L$SNM      000046         #5-62
L$SYL      000122         #5-62
L$TNM      000052         #5-62
L$TPT      000050         #5-62
L$TRB      000040         #5-62
L$TSZ      000070         #5-62
L$TYP      000063         #5-62
L$UNT      000003         #5-62
ME$CIR     000003          6-133
ME.FOR     177776          6-123
ME.FUN     177777          6-113
ME.MPR     177773          6-100
MF$SIG     177773          6-129
MO$COU     000060          6-114
MS.MOR     000002          6-136
NF.END   = 000200          6-158
R$$MPL   = ******          6-141
R$$11S   = ******          6-116
U.CW3      000014          6-158
$MBLUN   = ******   GX     6-154
$SAVRG   = ******   GX     6-94
```

```
                                                               M  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11    PAGE 7
SYMBOL CROSS REFERENCE                                CREF    04.00
SYMBOL  VALUE          REFERENCES
NMTACT    001222 R     #30-1106
NODACT    000230 R     #13-652
NODNAM    000774 R     #9-287
NODOWN    000760 R     #9-284
NUMACT    000774 R     #23-950
NXT$$$  = 000000       #8-82
N$$VCT  = ******        13-697
OWNACT    000256 R     #13-661
PAREND    000536 R      11-613     13-700     14-728   #16-787     17-816     18-835
PCKBCD    001104 R      22-932     23-958    #27-1031
PDVID   = ******  GX    13-689
PDVTA   = ******  GX    13-693
PLTACT    001176 R     #28-1070
PNAERR    000462 R      14-720    #15-750     17-812    19-859
PVAERR    000470 R      11-609     11-611     13-656    13-658     13-667     13-684     13-701    14-722     14-726
                       #15-753     17-814     20-889
RETACT    001066 R     #26-1009
R$PC    = ******  GX   *22-934    *23-960
R$$EIS  = ******        27-1046
R$$PRO  = ******        6-72       6-72
R$$11D  = ******        5-65       27-1046
R$$11M  = 000000        5-65
R$$11S  = ******        5-65       8-87       10-559    11-588
SACT      001022 R     #9-299
SALL      000462 R     #9-215
SALL1     000466 R     #9-218
SBBT      001126 R     #9-322
SBLK      001500 R     #9-413
SBSA      001214 R     #9-343
SBSD      001236 R     #9-350
SBSI      001260 R     #9-357
SCHN      001542 R     #9-422
SCMB      001564 R     #9-429
SCOS      000550 R     #9-238
SCUS      001606 R     #9-436
SDTE      001662 R     #9-446
SETCIR    000000 R     #8-82       10-556
SF.ACT  = 000200       #5-67
SF.ENA  = 000100       #5-67
SF.LPB  = 000004       #5-67
SF.MFL  = 000040       #5-67
SF.PAC  = 000020       #5-67
SF.REA  = 000010       #5-67
SF.SER  = 000001       #5-67
SF.SVC  = 000002       #5-67
SF.UNL  = 000040       #5-67
SHTM      000654 R     #9-261
SINA      001302 R     #9-364
SIND      001324 R     #9-371
SINI      001346 R     #9-378
SLCT      000626 R     #9-253
SLOC      001042 R     #9-306
```

```
                                                               N  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11    PAGE 8
```

```
                                          ; Circuits:
             000001                              LS$CHN =      1         ; Set PVC channel
             000002                              LS$CMB =      2         ; Set maximum data
             000004                              LS$CUS =      4         ; Set circuit usage
             000010                              LS$DTE =     10         ; Set DTE
             000020                              LS$MXR =     20         ; Set maximum recalls
             000040                              LS$MXW =     40         ; Set maximum window
             000100                              LS$NUM =    100         ; Set DLM number
             000200                              LS$RET =    200         ; Set recall timer
             001000                              LS$BLK =   1000         ; Set blocking
             100000                              LP$NXC = 100000         ; Circuit is a new X.25 circuit
                                          ; Lines:
             000400                              LS$HBT =    400         ; Set holdback timer
             000002                              LS$LMB = LS$CMB         ; Set max block
             000020                              LS$MRT = LS$MXR         ; Set max retransmits
             000040                              LS$MWN = LS$MXW         ; Set max window
             000200                              LS$NTI = LS$RET         ; Set retransmit timer
             010000                              LS$PRO =  10000         ; Set line protocol
                                          ; Common:
             020000                              LS$LCT =  20000         ; Set counter timer
                                          ; Groupings:
             020053                              LS$PVC = LS$CHN!LS$CMB!LS$DTE!LS$MXW!LS$LCT
             000362                              LS$SVC = LS$CMB!LS$MXR!LS$MXW!LS$NUM!LS$RET
             001004                              LS$DLM = LS$BLK!LS$CUS
```

```
                                                                  M 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-5
Symbol table

MS$TRI  000005          MX$CNM  000006          N$RTMX  000014          SETTH2= ****** GX       T$NRNI  000042
MS$UNR  000005          MX$CON  000006          N$RTM1  000014          SETTH3= ****** GX       T$NRPL  000005
MS$VMS  000004          MX$DAC  000020          N$RTM2  000015          SETUP   000476R         T$NRUL  000007
MS.DON  177600          MX$DTE  000020          N$RT1   000000          SETXMT= ****** GX       T$NVR   000001
MS.MOR  000002          MX$FIL  000034          N$RT2   000006          SF$ANM  000001          T$RPRI  000040
MS.PAR  000003          MX$LGC  000777          N$TCTL  000112          SF$DNM  000002          T$SVC   000034
MS.SUC  000001          MX$LIN  000020          N$TLC   000100          SF.ACT= 000200          T$T5    000030
MT$AR4  000003          MX$LON  000006          N$TNC   000114          SF.ENA= 000100          T$T6    000032
MT$ASC  000100          MX$NMS  000454          N$TRC   000106          SF.LPB= 000004          T$$KMG= 000000
MT$BIL  000001          MX$NOD  000006          N$TTCB  000110          SF.MFL= 000040          T$$MIN= 000000
MT$BYS  000011          MX$OBJ  000006          N$VER   000066          SF.PAC= 000020          V$$CTR= 001000
MT$CI   000007          MX$OWN  000040          N$XLEN  000124          SF.REA= 000010          X$$DBT= 000000
MT$CLE  000077          MX$PAR  000036          N$$ACC= 000001          SF.SER= 000001          X$$TLC= 000001
MT$COD  000200          MX$PAS  000010          N$$BUF= 000001          SF.SVC= 000002          ZF.COU= 001000
MT$CON  000001          MX$RAC  000047          N$$LDV= 000001          SF.UNL= 000040          ZF.DDM= 000001
MT$DMC  000004          MX$RID  000047          N$$MCP= 000001          SLTMA = ****** GX       ZF.DIA= 004000
MT$ETH  000006          MX$RPS  000047          N$$MLL= 000001          S$$WRG= 000000          ZF.DLC= 000002
MT$HEX  000040          MX$SID  000040          N$$MOV= 000010          S$$YSZ= 007600          ZF.DVP= 100000
MT$LPB  000005          MX$SNK  000377          N$$NCT= 000001          S.COST  000001          ZF.INI= 040000
MT$MAX  000037          MX$TYP  000077          N$$PEM= 000001          S.FLG   000000          ZF.KMX= 000020
MT$MUL  000100          MX$UID  000020          PDVTA = ****** GX       S.LEN   000004          ZF.LLC= 000004
MT$NLE  000017          M$$CRB= 000124          P$$P45= 000000          S.NMST  000002          ZF.LMC= 000100
MT$NON  000001          M$$CRX= 000000          P$$WRD= 000000          S.OWNR  000003          ZF.MAN= 020000
MT$NR4  000005          M$$FCS= 000000          Q$$OPT= 000010          T$FLAG  000044          ZF.MFL= 000010
MT$NTY  000060          M$$MGE= 000000          R$$DER= 000000          T$LIF   000013          ZF.MTM= 000400
MT$OCT  000060          M$$NET= 000000          R$$K11= 000001          T$LIFL  000013          ZF.MUX= 000040
MT$PHA  000002          M$$GVR= 000000          R$$SND= 000000          T$LIFO  000013          ZF.PSE= 002000
MT$POI  000000          NM.CLN  003400          R$$11M= 000000          T$LIFS  000013          ZF.SLI= 010000
MT$QP2  000010          NM.INI  002400          SETACT= ****** GX       T$LIN   000000          ZF.TIM= 000200
MT$ROU  000000          NM.OPP  003000          SETBBT= ****** GX       T$LIPS  000006          ZF.X3P= 000000
MT$RO4  000004          NM.VR2  002401          SETBSA= ****** GX       T$LLD   000012          ZS.ASN= 100000
MT$SEC  000000          NM.VR3  002402          SETBSD= ****** GX       T$LLDC  000045          ZS.BSY= 140000
MT$SGD  000020          NXTCIR= ****** GX       SETBSI= ****** GX       T$LLDL  000012          Z.AVL   000014
MT$SYS  000002          N$ADJ1  000072          SETCOS= ****** GX       T$LLDO  000012          Z.DAT   000016
MT$TER  000001          N$ADJ2  000074          SETHTM= ****** GX       T$LLDS  000012          Z.DSP   000000
MT$TRI  000002          N$CACH  000062          SETINA= ****** GX       T$LLEN  000046          Z.FLG   000010
MT$TYP  007777          N$CRC   000120          SETIND= ****** GX       T$LOPR  000002          Z.LEN = 000016
MT$USD  000000          N$HC1   000052          SETINI= ****** GX       T$LTCL  000024          Z.LLN   000006
MT$X25  000003          N$HC2   000056          SETMXB= ****** GX       T$LTIM  000026          Z.MAP   000020
MU$INC  000001          N$LV1   000002          SETNMT= ****** GX       T$LTPR  000014          Z.NAM   000004
MU$OUT  000002          N$LV2   000010          SETNTL= ****** GX       T$LTPS  000020          Z.PCB   000012
MU$PER  000000          N$MHC1  000036          SETOWN= ****** GX       T$NAPL  000004          Z.SCH   000007
MV$III  000000          N$MHC2  000044          SETSER= ****** GX       T$NFE   000000          $ADPVC= ****** GX
MV$IV   000001          N$PLD   000016          SETSTA= ****** GX       T$NLEN  000010          $SAVRG= ****** GX
MX$ACT  000020          N$PRI   000076          SETTAD= ****** GX       T$NNUL  000002          $$$SYS= 004374
MX$CIR  000020          N$ROA1  000022          SETTH1= ****** GX       T$NOPL  000006          $$$SYX= 000000
MX$CLN  000020          N$ROA2  000030

. ABS.  177776    000   (RW,I,GBL,ABS,OVR)
        001004    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 282
Work  file writes: 185
```

```
                                                                  N 3
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-6
```

```
                .TITLE  CICHPD - Circuit Change Parameters Operate (DECnet)
                .IDENT  /V05.00/
                .NLIST  BEX
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CIRCUIT CHANGE PARAMETERS OPERATE ROUTINE
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  21-Feb-83                 TMD [EDSSEG]
;       Split from CICHOP (see history in CIHOP) to permit heavier overlaying
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
;

```

```
                                                .SBTTL  SETSTA - SET A NEW CIRCUIT STATE
                                        ;+
                                        ; **-SETSTA-SET A NEW CIRCUIT STATE
                                        ;
                                        ; THIS ROUTINE IS CALLED TO SET A NEW CIRCUIT STATE
                                        ;
                                        ; INPUTS:
                                        ;       R4 = ADDRESS OF CONTEXT BLOCK WITH:
                                        ;            L$STA(R4) = THE DESIRED CIRCUIT STATE
                                        ;
                                        ; OUTPUTS:
                                        ;       IF CC, THE STATE WAS CHANGED,
                                        ;       ELSE, ERROR CODE IN R0
                                        ;
                                        ; REGISTERS:
                                        ;       R4, R5 ARE PRESERVED
                                        ;-

000734                                  SETSTA::
                                                .IF NDF R$$11S
000734  022764  177777  000050                  CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
000742  001007                                  BNE     10$                     ; BR IF NO
000744  012700  177752                          MOV     #ME.PNA,R0              ; INDICATE INVALID PARAMETER
000750  012764  000000  000072                  MOV     #MP$STA,L$PAR(R4)       ; ...
000756  000167  000546                          JMP     170$                    ; AND EXIT
                                                .ENDC
                                        ;
                                        ; POINT TO THE NETWORK MANAGENEMT WORD IN THE SLT
                                        ;
000762  016402  000042                  10$:    MOV     L$SLT(R4),R2            ; POINT TO THE SLT
000766  012700  177765                          MOV     #ME.CST,R0              ; ASSUME CIRCUIT IS IN WRONG STATE
000772  012764  000003  000072                  MOV     #ME$CIR,L$PAR(R4)       ; ...
001000  032762  040000  000000                  BIT     #LF.RDY,L.FLG(R2)       ; IS THE CIRCUIT LOADED ?
001006  001010                                  BNE     30$                     ; IF NE, YES - CONTINUE

001010  126427  000100  000001                  CMPB    L$STA(R4),#MS$OFF       ; Turning circuit off ?
001016  001402                                  BEQ     20$                     ; If EQ, yes, no-op
001020  000167  000504                          JMP     170$                    ; Else, no, error
001024  000167  000502                  20$:    JMP     180$                    ; No-op

001030  105762  000014                  30$:    TSTB    L.NSTA(R2)              ; MULTIPOINT CIRCUIT OR ETHERNET ?
001034  001015                                  BNE     50$                     ; IF NE, YES
001036  032762  100000  000000                  BIT     #LF.ACT,L.FLG(R2)       ; IS THE CIRCUIT OFF?
001044  001006                                  BNE     40$                     ; BR IF NO
001046  126427  000100  000001                  CMPB    L$STA(R4),#MS$OFF       ; TURNING CIRCUIT OFF?
001054  001002                                  BNE     40$                     ; BR IF NE
001056  000167  000450                          JMP     180$                    ; No-op
001062  062702  000020                  40$:    ADD     #L.NMST,R2              ; ELSE, POINT TO LINE MANAGEMENT WORD
001066  000432                                  BR      80$                     ; ...
001070  012700  177767                  50$:    MOV     #ME.IID,R0              ; ASSUME INVALID CIRCUIT-ID
001074  032762  000400  000000                  BIT     #LF.BRO,L.FLG(R2)       ; ETHERNET CIRCUIT ?
001102  001006                                  BNE     60$                     ; IF NE YES - NO TRIB REQUIRED
001104  132764  000002  000041                  BITB    #LP$TRB,L$PFG(R4)       ; ELSE, WAS A TRIB EXPLICITLY GIVEN ?
001112  001002                                  BNE     60$                     ; IF NE, yes - OK
001114  000167  000410                          JMP     170$                    ; ELSE INVALID CIRCUIT-ID
001120  016402  000050                  60$:    MOV     L$TPT(R4),R2            ; POINT TO TRIB FLAGS
001124  132762  000200  000000                  BITB    #SF.ACT,S.FLG(R2)       ; IS THE CIRCUIT OFF?
```

```
    828
    829                                          .DSABL  LSB
    830
    831                                          .IF NDF R$$11S
```

```
S.OWNR  000003        S4.HHT= 000040        T$NNUL  000002        T.TIO   000057        US.BSY= 000200
S1.DEC= 001000        S4.HSY= 000200        T$NOPL  000006        T.TKSZ  000060        US.FOR= 000040
S1.DPR= 000400        S4.RGS= 020000        T$NRNI  000042        T.TOD   000070        US.FRK= 000002
S1.DSI= 004000        S4.SFC= 040000        T$NRPL  000005        T.TRLG= 000076        US.KPF= 000001
S1.ESC= 000002        S4.VFL= 000010        T$NRUL  000007        T.TRLN  000005        US.LAB= 000004
S1.IBF= 002000        S5.ABP= 100000        T$NVR   000001        T.TRPN  000004        US.MDE= 000002
S1.IBY= 000200        S5.BCC= 020000        T$RPRI  000040        T.UCB   000026        US.MDM= 000020
S1.OBY= 000100        S5.DAO= 040000        T$SVC   000034        T.XASQ  000010        US.MNT= 000100
S1.PTH= 000010        S5.HPC= 000014        T$T5    000030        T.XBTC  000024        US.OFL= 000001
S1.RES= 010000        S5.HPO= 000020        T$T6    000032        T.XDAT  000030        US.PUB= 000004
S1.RNE= 000020        S5.ITI= 000100        T$$KMG= 000000        T2.ABO= 000100        US.PWF= 000010
S1.RNF= 020000        S5.OXF= 000040        T$$MIN= 000000        T2.AST= 100000        US.RED= 000002
S1.RSP= 000004        S5.RPO= 002000        T.ACTL  000052        T2.CAF= 000400        US.SHR= 000001
S1.RST= 000001        S5.SW1= 000001        T.ASTL  000016        T2.CHK= 020000        US.SPU= 000002
S1.TNE= 040000        S5.TMM= 000002        T.ATT   000062        T2.CKD= 010000        US.UMD= 000010
S1.TSY= 000040        S5.VER= 010000        T.BTTC  000074        T2.DST= 040000        US.VV = 000001
S1.USI= 100000        S5.XOF= 000004        T.BUF   000006        T2.FXD= 002000        US.WCK= 000010
S2.BEL= 020000        S5.XON= 000010        T.CPCB  000004        T2.HLT= 000200        UU.ABO= 000400
S2.BRQ= 000020        S6.EIO= 000400        T.DELT  000056        T2.REX= 001000        UU.ATN= 000100
S2.CR = 002000        S6.RDI= 100000        T.DIN   000044        T2.SEF= 004000        UU.AVN= 000004
S2.CTO= 040000        S6.RLU= 001000        T.DOUT  000042        T2.SPN= 000004        UU.GUS= 000010
S2.CTS= 100000        TH1TBL  000252R       T.DPRI  000040        T2.STP= 000020        UU.IOS= 002000
S2.ELF= 001000        TH2TBL  000264R       T.EFLG  000022        T2.WFR= 000001        UU.ONL= 000020
S2.FLF= 000400        TH3TBL  000276R       T.ERRC  000020        T3.ACP= 100000        UU.RCT= 000002
S2.IRQ= 000200        TS.BLK= 171700        T.IOC   000003        T3.CAL= 000100        UU.RDY= 000200
S2.OBF= 004000        TS.CKP= 000200        T.LBF   000046        T3.CLI= 001000        UU.SER= 000001
S2.ORQ= 000100        TS.CKR= 000100        T.LBN   000041        T3.GFL= 000010        UU.SIO= 001000
S2.PCU= 010000        TS.EXE= 100000        T.LDV   000044        T3.MCR= 004000        UU.SPC= 000040
S2.RCU= 000001        TS.HLD= 002000        T.LNK   000000        T3.NET= 000020        U.ACB   000062
S2.SRQ= 000040        TS.MSG= 020000        T.LTM   000054        T3.NSD= 000200        U.ACP = 000032
S2.WAL= 000010        TS.NRP= 010000        T.MXSZ  000050        T3.PMD= 040000        U.ADMA  000066
S2.WRA= 000006        TS.OUT= 000400        T.NAM   000006        T3.PRV= 010000        U.AFLG  000064
S2.WRB= 000002        TS.RDN= 040000        T.NDM   000066        T3.REM= 020000        U.ATT   000022
S3.ACR= 000001        TS.RUN= 004000        T.OFF   000066        T3.ROV= 000040        U.BPKT= 000044
S3.CTC= 000004        TS.STP= 001000        T.PCB   000046        T3.RST= 000400        U.BUF   000024
S3.FDX= 000200        TWKLIN  002116RG      T.PIND  000007        T3.SLV= 002000        U.CBF = 000032
S3.ICE= 001000        T$FLAG  000044        T.PRI   000002        T3.SWS= 000002        U.CLI   177772
S3.MHE= 000400        T$LIF   000013        T.QRAC  000060        UC.ALG= 000200        U.CNT   000030
S3.NEC= 000020        T$LIFL  000013        T.QRIN  000062        UC.ATT= 000010        U.COTQ  000030
S3.PPT= 020000        T$LIFO  000013        T.QRPD  000064        UC.KIL= 000004        U.CTCB  000026
S3.PTH= 004000        T$LIFS  000013        T.RBF   000050        UC.LGH= 000003        U.CTL   000004
S3.RAL= 000010        T$LIN   000000        T.RBTC  000020        UC.NPR= 000100        U.CTYP  000052
S3.RES= 010000        T$LIPS  000006        T.RCVL  000012        UC.PWF= 000020        U.CW1   000010
S3.RUB= 040000        T$LLD   000012        T.RDAT  000034        UC.QUE= 000040        U.CW2   000012
S3.TAB= 000002        T$LLDC  000045        T.RRFL  000072        UD.UNS= 000000        U.CW3   000014
S3.TME= 002000        T$LLDL  000012        T.RTM   000055        UD.160= 000004        U.CW4   000016
S3.TSY= 000040        T$LLDO  000012        T.SAST  000054        UD.200= 000001        U.CYL = 000104
S3.8BC= 000100        T$LLDS  000012        T.SEL   000040        UD.556= 000002        U.DCB   000000
S4.ABD= 100000        T$LLEN  000046        T.SELT  000052        UD.625= 000005        U.FCDE= 000042
S4.ANI= 000400        T$LOPR  000002        T.SFNC  000014        UD.8K = 000006        U.FNUM= 000040
S4.AVO= 001000        T$LTCL  000024        T.SITC  000072        UD.800= 000003        U.GRP = 000102
S4.BLK= 002000        T$LTIM  000026        T.SRCT  000071        UM.CLI= 000036        U.KCSR= 000032
S4.DEC= 004000        T$LTPR  000014        T.STAT  000002        UM.DSB= 000200        U.KCS6= 000034
S4.DLO= 000100        T$LTPS  000020        T.ST2   000034        UM.NBR= 000400        U.LUIC  177774
S4.EDT= 010000        T$NAPL  000004        T.ST3   000036        UM.OVR= 000001        U.MEDI= 000040
S4.HFF= 000020        T$NFE   000000        T.TCBL  000030        US.ABO= 000001        U.MLUN= 000050
S4.HFL= 000007        T$NLEN  000010        T.TIMC  000016        US.BSP= 000002        U.MUP   177772
```

```
                                                             M 8
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14
Table of contents

     5-   53     MACRO CALLS AND LOCAL DATA
     6-  102     Context area definition
     7-  107     Control block parameter format tables
     8-  134     CICHPX - Set X.25 circuit parameters
     9-  176     SETPVC - Set PVC parameters
    10-  202     SETSVC - Set SVC parameters
    11-  246     CKCSTA - Check circuit is in STATE OFF
    12-  300     SETCTR - Set flags in Line Counter block
    13-  321     SVCFLG - Set flags in SVC block
    14-  348     CKDSVC - Check for DLM SVC
    15-  373     SETLCT - SET COUNTER TIMER
    16-  411     $ADPVC - Create a new PVC
```

```
                                                             N 8
CICHPX - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:14  Page 4
```

```
                                                .SBTTL  SVCFLG - Set flags in SVC block
                                        ;+
                                        ;*** SVCFLG - Set flags in SVC block
                                        ;
                                        ; INPUTS
                                        ;       R4 - Address of context area
                                        ;       R5 - Address of SVC descriptor (mapped)
                                        ;
                                        ; OUTPUTS
                                        ;       Blocking and Incoming SVC flags set as required
                                        ;-

000720                                  SVCFLG:
000720  032764  000004  000076                  BIT     #LS$CUS,L$FLX(R4)       ;; Is USAGE change specified?
000726  001412                                  BEQ     10$                     ;; Branch if USAGE unchanged
000730  042765  000002  000010                  BIC     #SF$INC,S$FLG(R5)       ;; Assume OUTGOING circuit
000736  022764  000001  000260                  CMP     #MU$INC,L$CUS(R4)       ;; Was INCOMING specified?
000744  001003                                  BNE     10$                     ;; If not, we were right
000746  052765  000002  000010                  BIS     #SF$INC,S$FLG(R5)       ;; ... otherwise reset INCOMING flag
000754  032764  001000  000076          10$:    BIT     #LS$BLK,L$FLX(R4)       ;; Was BLOCKING change requested?
000762  001412                                  BEQ     20$                     ;; Branch if no change
000764  042765  000001  000010                  BIC     #SF$BLK,S$FLG(R5)       ;; Assume blocking disabled
000772  022764  000000  000262                  CMP     #MB$ENA,L$BLK(R4)       ;; Was ENABLED requested?
001000  001003                                  BNE     20$                     ;; If not, right again
001002  052765  000001  000010                  BIS     #SF$BLK,S$FLG(R5)       ;; Else set ENABLED flag
001010                                  20$:    RETURN
```

```
                                                                 M 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 5
SYMBOL CROSS REFERENCE                                  CREF   04.00

SYMBOL   VALUE              REFERENCES

L.NSTA    000014            #5-72
L.OWNR    000021            #5-72
L.UNT     000013            #5-72
MB$ENA    000000             12-316     13-343
ME$CIR    000003             10-235     11-294
ME.CST    177765             10-234     11-288
ME.GRO    177745             8-164      9-198      10-241
ME.IID    177767             11-293
ME.PNA    177752             8-168      15-406
ME.PVA    177760             8-172
MP$BLK    001616             8-169
MP$CUS    002127             8-173
MP$LCT    000156             15-407
MS.MOR    000002             8-146
MU$INC    000001             13-337
MU$PER    000000             8-157
N$$VCT  = ******             10-218     10-227     10-238     14-362     14-369     14-370     15-391     15-392     15-400
                             15-401     15-403
PSIPT   = ******   GX        15-396
PX$BLK  = 000040             12-315     12-318
PX$DLM  = 000200             14-365
PX$SVC  = 000100             14-367
R$PC    = 000016            #5-100
R$R0    = 000002            #5-94      *10-236
R$R1    = 000004            #5-95
R$R2    = 000006            #5-96
R$R3    = 000010            #5-97
R$R4    = 000012            #5-98
R$R5    = 000014            #5-99
R$$PRO  = ******             6-104      6-104
R$$11S  = ******             4-1
SETCTR    000662 R           10-225    #12-312
SETLCT    001070 R           9-195     #15-386
SETPVC    000326 R           8-150     #9-187
SETSVC    000400 R           8-162     #10-213
SF$BLK  = 000001             13-342     13-345
SF$INC  = 000002             13-336     13-339
SF.ACT  = 000200            #5-72       11-281
SF.ENA  = 000100            #5-72
SF.LPB  = 000004            #5-72
SF.MFL  = 000040            #5-72
SF.PAC  = 000020            #5-72
SF.REA  = 000010            #5-72
SF.SER  = 000001            #5-72
SF.SVC  = 000002            #5-72
SF.UNL  = 000040            #5-72
SVCFLG    000720 R           10-228    #13-333
SVCMAP    000000 R          #7-122      10-230
S$ADSZ    000007             7-125      7-126
S$DTE     000020             7-130
S$FLG     000010            *13-336    *13-339    *13-342    *13-345
S$LEN     000032             7-131
```

```
                                                                 N 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15     PAGE 6
```

```
    274 000654  110223                          MOVB    R2,(R3)+                ; STORE USAGE
    275                                 ;
    276                                 ; RETURN LINE TYPE INDICATOR
    277                                 ;
    278 000656  026727  177120  074361  60$:    CMP     OWNER,#^RSNA            ; is this an SNA circuit ?
    279 000664  001456                          BEQ     75$                     ; if EQ, yes- not applicable
    280 000666  004567  000000G                 JSR     R5,SETPAR               ; SET LINE TYPE PARAMETER
    281 000672  002130                          .WORD   MP$TYP                  ; ...
    282 000674  000201                          .WORD   MT$COD!1                ; ONE CODED BYTE
    283 000676  112702  000003                  MOVB    #MT$X25,R2              ; ASSUME X25
    284 000702  022764  015355  000000          CMP     #^RDLM,L$DDM(R4)        ; IS THIS DLM?
    285 000710  001423                          BEQ     65$                     ; BR IF YES
    286 000712  112702  000006                  MOVB    #MT$ETH,R2              ; ASSUME ETHERNET
    287 000716  032760  000400  000000          BIT     #LF.BRO,L.FLG(R0)       ; BROADCAST CHANNEL ?
    288 000724  001015                          BNE     65$                     ; IF NE, YES
    289 000726  112702  000000                  MOVB    #MT$POI,R2              ; ASSUME POINT-TO-POINT
    290 000732  032760  000020  000000          BIT     #LF.MTP,L.FLG(R0)       ; MULTIPOINT LINE ?
    291 000740  001407                          BEQ     65$                     ; IF EQ, NO
    292 000742  112702  000001                  MOVB    #MT$CON,R2              ; ELSE ASSUME MULTIPOINT CONTROLLER
    293 000746  105760  000014                  TSTB    L.NSTA(R0)              ; MULTIPOINT CONTROLLER ?
    294 000752  001002                          BNE     65$                     ; IF NE, YES
    295 000754  112702  000002                  MOVB    #MT$TRI,R2              ; ELSE, MULTIPOINT TRIBUTARY
    296 000760  110223                  65$:    MOVB    R2,(R3)+                ; SET LINE TYPE BYTE IN MESSAGE
    297                                 ;
    298                                 ; RETURN DTE,MAXIMUM DATA, MAXIMUM WINDOW
    299                                 ;
    300
    301                                         .IF NDF R$$PRO  ; PRO/DECNET
    302
    303 000762  026427  000000  015355          CMP     L$DDM(R4),#^RDLM        ; IS THIS DLM?
    304 000770  001014                          BNE     75$                     ; BR IF NO
    305 000772  032767  000001  177020          BIT     #FL.SVC,FLAG            ; IS THIS AN SVC?
    306 001000  001403                          BEQ     70$                     ; BR IF NO
    307 001002                                  CALL    SVCBLK                  ; GET SVC BLOCK INFO
    308 001006  000413                          BR      80$                     ; CONTINUE
    309 001010  005000                  70$:    CLR     R0                      ; Get PLI port number (no sign extend)
    310 001012  156700  176772                  BISB    PVCPRT,R0
    311 001016                                  CALLR   PVCBLK                  ; Get PVC block information and exit
    312
    313                                         .IFTF   ; NDF R$$PRO
    314
    315 001022                          75$:
    316                                         .IFT    ; NDF R$$PRO
    317
    318                                           .IF NDF S$$BAS & R$$RTR
    319
    320 001022  032760  000020  000000          BIT     #LF.MTP,L.FLG(R0)       ; MULTI-POINT LINE ?
    321 001030  001402                          BEQ     80$                     ; IF EQ, NO - SKIP PARAMETERS
    322 001032                                  CALL    MTPCHA                  ; get multipoint characteristics
    323
    324                                           .ENDC
    325
    326                                         .ENDC   ; NDF R$$PRO
    327
    328
    329 001036  042764  040000  000064  80$:    BIC     #LF$MLT,L$FLG(R4)       ; Assume no multiple adjacencies
    330 001044  032764  100000  000064          BIT     #LF$SEG,L$FLG(R4)       ; Partial response in progress?
```

```
                                .SBTTL  SVCBLK - Get SVC block parameters

                        ;+
                        ;
                        ; SVCBLK - GET SVC BLOCK PARAMETERS
                        ;
                        ; INPUTS:
                        ;       R3 - ADDRESS OF NEXT AVAILABLE BYTE IN OUTPUT BUFFER
                        ;       R4 - ADDRESS OF CONTEXT AREA
                        ;       SVCMAP - APR BIAS FOR SVC DESCRIPTOR
                        ;       SVCADD - APR 6 VIRTUAL ADDRESS OF SVC DESCRIPTOR
                        ;
                        ; OUTPUTS:
                        ;       R3 - UPDATED
                        ;
                        ;-
003114                  SVCBLK: SWSTK$  20$                             ;; ENTER SYSTEM STATE
003120                          SAVMAP                                  ;; SAVE CURRENT MAPPING
003124                          MAP     SVCMAP                          ;; MAP TO SVC DESCRIPTOR
003132  016700  174650          MOV     SVCADD,R0                       ;; GET ADDRESS OF DESCRIPTOR
                        ;
                        ; FORMAT DTE ADDRESS
                        ;
003136  116002  000007          MOVB    S$ADSZ(R0),R2                   ;; GET LENGTH OF DTE ADDRESS
003142  001411                  BEQ     10$                             ;; BR IF INCOMING SVC
003144  004567  000000G         JSR     R5,SETPAR                       ;; Set NUMBER parameter           ;[TD001]
003150  001642                  .WORD   MP$NUM                          ;; ...                            ;[TD001]
003152  000100                  .WORD   MT$ASC                          ;; ASCII IMAGE FIELD
003154  010001                  MOV     R0,R1                           ;; POINT TO SVC DESCRIPTOR
003156  062701  000020          ADD     #S$DTE,R1                       ;; POINT TO DTE ADDRESS
003162                          CALL    BCDASC                          ;; STORE ASCII DTE ADDRESS
                        ;
                        ; FORMAT MAXIMUM DATA
                        ;
003166  004567  000000G 10$:    JSR     R5,SETPAR                       ;; SET MAX DATA PARAMATER
003172  002142                  .WORD   MP$CMB                          ;; ...
003174  000002                  .WORD   2                               ;; TWO BYTE DECIMAL FIELD
003176  116023  000004          MOVB    S$PKSZ(R0),(R3)+                ;; STORE MAX DATA
003202  116023  000005          MOVB    S$PKSZ+1(R0),(R3)+              ;; ...
                        ;
                        ; FORMAT MAXIMUM WINDOW
                        ;
003206  004567  000000G         JSR     R5,SETPAR                       ;; SET MAX WINDOW
003212  002143                  .WORD   MP$MXW                          ;; ...
003214  000001                  .WORD   1                               ;; ONE BYTE DECIMAL FIELD
003216  116023  000006          MOVB    S$WND(R0),(R3)+                 ;; STORE MAX WINDOW
003222                          RESMAP                                  ;; RESTORE PREVIOUS MAPPING
003226  010366  000010          MOV     R3,10(SP)                       ;; STORE UPDATED R3
003232                  20$:    RETURN

```

```
001]
001]
```

```
                                                                         M 13
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 5

SYMBOL CROSS REFERENCE                                   CREF   04.00

SYMBOL   VALUE               REFERENCES

MT$ASC     000100              9-251     16-698     19-864
MT$COD     000200              9-233     9-273      9-282      16-685     16-691     20-904
MT$CON     000001              9-292
MT$ETH     000006              9-286
MT$MUL     000100              16-685    16-691
MT$POI     000000              9-289
MT$TRI     000002              9-295
MT$USD     000000              9-201     9-205
MT$X25     000003              9-283
MU$INC     000001              9-267
MU$OUT     000002              9-270
MU$PER     000000              9-264
M$$MGE   = 000000              6-74      6-74       14-532     14-537     17-765     17-782
NODE       000000 R           #8-89      9-156     *14-535    *14-553    *14-571
N$ADJ1     000072              14-531    15-624
N$ADJ2     000[illegible]              14-536    15-639
N$PLD      00001[illegible]              10-358    10-362
N$TLC      000100              17-752    17-760     17-761
N$$VCT   = ******              10-366    11-385     11-396     11-397     11-410     11-418     13-473     13-476     13-483
                               14-530    14-551     14-578     15-623     15-624     15-639     15-643     17-750     17-760
                               17-783    17-785     17-791     18-818     18-819     18-833     19-854     19-855     19-883
OWNER      000002 R           #8-90     *9-256      9-278
PDVID    = ******    GX        10-351    14-515
PDVTA    = ******    GX        9-252     10-355     14-519     17-738
PS$UP    = 000012              9-164
PT$BRO   = 000200              9-166     9-196      14-541
PT$DRT   = 000100              9-172
PVCBLK   = ******    GX        9-311
PVCCHA   = ******    GX        9-131
PVCPRT     000010 R           #8-93      9-310     *17-795
PX$BLK   = 000040              13-480
PX$DLM   = 000200              17-773
PX$SVC   = 000100              17-775
P$DRTR     000036              9-168     9-171
P$LST      000000              9-164
P$NRNI     000040              9-202
P$RPRI     000042              9-206
P$TYP      000001              9-166     9-170      9-196      14-541
RBIAS    = 000002             #6-79
RX.NXM   = 000020             #6-74
RX.OVR   = 000010             #6-74
R$R1     = ******    GX       *14-538   *14-552    *14-57[illegible]
R$R2     = ******    GX       *10-363   *11-388    *11-392
R$R3     = ******    GX       *11-419
R$$EIS   = ******              14-588
R$$PRO   = ******              9-129     9-301      10-369
R$$RTR   = ******              9-318     10-369
R$$11D   = ******              6-69      14-588
R$$11M   = 000000              6-69      14-532     14-537     17-765     17-782
R$$11S   = ******              6-69
R.CSR      000056             #6-74
R.FLG      000064             #6-74
```

```
                                                                         N 13
CIRECH     CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 6
```

```
                                                                                    M 14
CIRECO - READ/AND OR ZERO CIRCU MACRO V05.03b Saturday 29-Jun-85 12:17  Page 8
LOCAL DATA

     72                                             .SBTTL  LOCAL DATA
     73
     74                                     ;
     75                                     ; LOCAL DATA
     76                                     ;
     77
     78 000000                              SCRATC: .BLKW   1                           ; SCRATCH AREA
```

```
**FILE**ID**CIREIN




  CCCCCCCC     IIIIII     RRRRRRRR      EEEEEEEEEE     IIIIII     NN      NN
  CCCCCCCC     IIIIII     RRRRRRRR      EEEEEEEEEE     IIIIII     NN      NN
CC               II       RR      RR    EE               II       NN      NN
CC               II       RR      RR    EE               II       NN      NN
CC               II       RR      RR    EE               II       NNNN    NN
CC               II       RR      RR    EE               II       NNNN    NN
CC               II       RRRRRRRR      EEEEEEEE         II       NN  NN  NN
CC               II       RRRRRRRR      EEEEEEEE         II       NN  NN  NN
CC               II       RR  RR        EE               II       NN    NNNN
CC               II       RR  RR        EE               II       NN    NNNN
CC               II       RR    RR      EE               II       NN      NN        ....
CC               II       RR    RR      EE               II       NN      NN        ....
  CCCCCCCC     IIIIII     RR      RR    EEEEEEEEEE     IIIIII     NN      NN        ....
  CCCCCCCC     IIIIII     RR      RR    EEEEEEEEEE     IIIIII     NN      NN        ....




LL             SSSSSSSS    TTTTTTTTTT
LL             SSSSSSSS    TTTTTTTTTT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL           SS                TT
LL             SSSSSS          TT
LL             SSSSSS          TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LL                   SS        TT
LLLLLLLLLL   SSSSSSSS          TT
LLLLLLLLLL   SSSSSSSS          TT
```

```
CIREIN      CREATED BY  MACRO  ON 29-JUN-85 AT 12:17      PAGE 3

MACRO CROSS REFERENCE                                     CREF   04.00

MACRO NAME        REFERENCES

CALL               6-134
CIRCX$            #5-59          5-62
HDRDF$            #5-59          5-63
MANDF$            #5-59          5-64
NSFDF$            #5-59          5-65
RESMAP            #5-60
RESRG             #5-59
RETURN             6-171
SAVMAP            #5-60
SAVRG             #5-59
SOB                6-109
UCBDF$            #5-59          5-6
```

```
                                                                         N  1
CICHIN     CREATED BY  MACRO  ON 29-JUN-85 AT 12:11       PAGE 8

SYMBOL CROSS REFERENCE                                    CREF   04.00

SYMBOL   VALUE               REFERENCES

SLTM       000702 R          #9-269
SMXB       001456 R          #9-406
SMXR       001710 R          #9-454
SMXW       001732 R          #9-461
SNMT       001150 R          #9-329
SNUM       001754 R          #9-468
SOWN       000720 R          #9-276
SPLT       001104 R          #9-315
SRET       002002 R          #9-476
SSER       000570 R          #9-245
SSTA       000474 R          #9-225
STAD       001002 R          #9-292
STH1       001370 R          #9-385
STH2       001412 R          #9-392
STH3       001434 R          #9-399
SXMT       001172 R          #9-336
S.COST     000001            #5-67
S.FLG      000000            #5-67
S.LEN      000004            #5-67
S.NMST     000002            #5-67
S.OWNR     000003            #5-67
TADACT     000420 R          #14-718
TH1ACT     001342 R          #38-1250
TH2ACT     001354 R          #39-1268
TH3ACT     001366 R          #40-1286
TR$BYT   = 000004            #8-82
TR$EOM   = 000010            #8-82
TR$ERR   = 000016            #8-82
TR$IMG   = 000000            #8-82
TR$MAT   = 000020            #8-82
TR$MBY   = 000006            #8-82
TR$NUL   = 000012            #8-82
TR$PAR   = 000002            #8-82
TR$SUB   = 000014            #8-82
XMTACT     001234 R          #31-1124
X$$MCB   = ******             5-65       5-65
ZF.COU   = 001000            #5-65
ZF.DDM   = 000001            #5-65
ZF.DIA   = 004000            #5-65
ZF.DLC   = 000002            #5-65
ZF.DVP   = 100000            #5-65
ZF.INI   = 040000            #5-65
ZF.KMX   = 000020            #5-65
ZF.LLC   = 000004            #5-65       13-695
ZF.LMC   = 000100            #5-65
ZF.MAN   = 020000            #5-65
ZF.MFL   = 000010            #5-65
ZF.MTM   = 000400            #5-65
ZF.MUX   = 000040            #5-65
ZF.PSE   = 002000            #5-65
ZF.SLI   = 010000            #5-65
ZF.TIM   = 000200            #5-65
```

```
                                        .SBTTL  DISPATCH TABLE FOR OPERATE ROUTINES

                                        .IF DF  R$$11S
                                        .PSECT  NIXCOD                  ; CODE MAY RESIDE IN APR 6
                                        .ENDC
                                ;
                                ; DISPATCH TABLE FOR OPERATE ROUTINES (L$FLG)
                                ;
000000                          DISTBL:
                                        .IF NDF R$$11S
                                        .IF NDF S$$BAS & R$$RTR
000000  000001  000000G                 .WORD   LS$NTL,SETNTL           ; NTL SET FUNCTIONS
                                        .IFTF   ; S$$BAS & R$$RTR
000004  000004  000000G                 .WORD   LS$COS,SETCOS           ; SET COST
000010  000010  000000G                 .WORD   LS$OWN,SETOWN           ; SET OWNER
                                        .IFT    ; S$$BAS & R$$RTR
000014  000020  000000G                 .WORD   LS$TAD,SETTAD           ; SET TRIBUTARY ADDRESS
000020  000040  000000G                 .WORD   LS$ACT,SETACT           ; SET MULTIPOINT ACTIVE RATIO
                                        .ENDC   ; NDF S$$BAS & R$$RTR
                                        .IFTF   ; NDF R$$11S
000024  000002  000000G                 .WORD   LS$STA,SETSTA           ; SET STATE
                                        .IFT    ; NDF R$$11S
                                        .IF NDF S$$BAS & R$$RTR & R$$PRO
000030  000200  000000G                 .WORD   LC$NTL,CLENTL           ; NTL CLEAR FUNCTION
000034  010000  000000G                 .WORD   LS$BBT,SETBBT           ; SET BABBLE TIMER
000040  020000  000000G                 .WORD   LS$NMT,SETNMT           ; SET NORMAL TIMER
000044  040000  000000G                 .WORD   LS$XMT,SETXMT           ; SET TRANSMIT TIMER
                                        .ENDC   ; NDF S$$BAS & R$$RTR & R$$PRO
000050  100000  000000G                 .WORD   LS$SER,SETSER           ; SET SERVICE
                                        .ENDC   ; NDF R$$11S
000054  000000                          .WORD   0                       ; END OF TABLE

                                        .IF NDF R$$11S
                                ;
                                ; DISPATCH TABLE FOR OPERATE ROUTINES (L$FL1)
                                ;
000056                          DISTB1: .IF NDF S$$BAS & R$$RTR & R$$PRO
000056  000001  000000G                 .WORD   LS$BSA,SETBSA           ; SET ACTIVE BASE
000062  000002  000000G                 .WORD   LS$BSD,SETBSD           ; SET DYING BASE
000066  000004  000000G                 .WORD   LS$BSI,SETBSI           ; SET INACTIVE BASE
000072  000010  000000G                 .WORD   LS$INA,SETINA           ; SET ACTIVE INCREMENT
000076  000020  000000G                 .WORD   LS$IND,SETIND           ; SET DYING INCREMENT
000102  000040  000000G                 .WORD   LS$INI,SETINI           ; SET INACTIVE INCREMENT
000106  000100  000000G                 .WORD   LS$TH1,SETTH1           ; SET DEAD THRESHOLD
000112  000200  000000G                 .WORD   LS$TH2,SETTH2           ; SET DYING THRESHOLD
000116  000400  000000G                 .WORD   LS$TH3,SETTH3           ; SET INACTIVE THRESHOLD
000122  001000  000000G                 .WORD   LS$MXB,SETMXB           ; SET MAXIMUM BLOCKS
                                        .ENDC   ; NDF S$$BAS & R$$RTR & R$$PRO

000126  010000  000000G                 .WORD   LS$HTM,SETHTM           ; SET HELLO TIMER
000132  000000                          .WORD   0                       ; END OF TABLE

                                        .ENDC   ; NDF R$$11S

```

```
CICHOP - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:12  Page 13-6
Symbol table

Size of work file: 36059 Words  ( 141 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:00:38.04
SY:CICHOP.V2,[135,134]CICHOP/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]V2,CICHOP
```

```
     49                                                 .SBTTL  MACRO CALLS AND LOCAL DATA
     50                                         ;
     51                                         ; MACRO LIBRARY CALLS
     52                                         ;
     53                                                 .MCALL  MANDF$,SAVRG,RESRG,NTLDF$,PDVDF$
     54                                                 .MCALL  MESAG$,SLTDF$,NSFDF$
     55                                                 .MCALL  ECDDB$,ALUN$S,WSIG$S,CEACC$
     56                                                 .MCALL  CCBDF$,MAP,BIAS,RETC,MAP,SAVMAP,RESMAP,CTRDF$
     57                                                 .MCALL  XPDDB$,UCBDF$,HDRDF$,PLBDF$,TCBDF$,DHBDF$
     58                                                 .MCALL  CICCX$,CALL,CALLR,CALLX
     59
     60                                                 .IF NDF R$$PRO  ; PRO/DECNET
     61
     62                                                 .MCALL  DMPDF$
     63
     64 000000                                          DMPDF$                          ; DEFINE DMP LINE TABLE OFFSETS
     65
     66                                                 .ENDC   ; NDF R$$PRO
     67
     68 000000                                          CTRDF$                          ; DEFINE COUNTER BLOCK OFFSETS
     69 000000                                          MANDF$                          ; DEFINE NETWORK MANAGEMENT SYMBOLS
     70 000000                                          NSFDF$                          ; DEFINE NS: FEATURES MASKS
     71 000000                                          NTLDF$                          ; DEFINE NETWORK LOADER SYMBOLS
     72 000000                                          PDVDF$                          ; DEFINE PDV OFFSETS
     73 000000                                          SLTDF$                          ; DEFINE SLT OFFSETS
     74 000000                                          ECDDB$                          ; DEFINE ECL'S  DDB SYMBOLS
     75 000000                                          XPDDB$                          ; DEFINE XPT'S DDB SYMBOLS
     76 000000                                          CCBDF$                          ; DEFINE CCB OFFSETS
     77 000000                                          HDRDF$                          ; DEFINE TASK HEADER OFFSETS
     78 000000                                          UCBDF$                          ; DEFINE UCB OFFSETS
     79 000000                                          PLBDF$                          ; DEFINE PLB OFFSETS
     80 000000                                          TCBDF$                          ; DEFINE TCB SYMBOLS
     81 000000                                          DHBDF$                          ; DEFINE DECNET HOME BLOCK SYMBOLS
     82
```

```
001132 001006                           BNE     70$                     ; BR IF NO
001134 126427  000100  000001           CMPB    L$STA(R4),#MS$OFF       ; TURNING CIRCUIT OFF?
001142 001002                           BNE     70$                     ; BR IF NO
001144 000167  000362                   JMP     180$                    ; ELSE, NOOP
001150 062702  000002           70$:    ADD     #S.NMST,R2              ; POINT TO TRIB MANAGEMENT WORD
                                ;
                                ; IF THE CIRCUIT OWNER IS TRANSPORT, MAKE SURE THE NODE IS ON
                                ;
001154 116203  000001           80$:    MOVB    1(R2),R3                ; GET THE OWNER'S PDV INDEX
001160 067703  000000G                  ADD     @PDVTA,R3               ; INDEX INTO PDV TABLE           ; RJK08
001164 011303                           MOV     (R3),R3                 ; POINT TO PDV
001166 026327  000004  114224           CMP     Z.NAM(R3),#^RXPT        ; IS THE OWNER TRANSPORT ?
001174 001121                           BNE     130$                    ; IF NE, NO - BYPASS STATE CHECK

001176                                  CALL    DEALUN                  ; ELSE, DEASSIGN THE TEMPORARY LUN
001202                                  ALUN$S  #$TMLUN,#"NS            ; ASSIGN THE LUN TO NS:
001222 103500                           BCS     110$                    ; IF CS, NETWORK IN WRONG STATE

                                        .IF     DF,R$$MPL

                                        SWSTK$  85$
                                        SAVMAP                          ;; SAVE CURRENT KISAR6 MAPPING
                                        MOV     @SAHDB,@KSAR6           ;; MAP TO MY EXTERNAL HEADER
                                        MOV     @SAHPT,R1               ;; POINT TO THE HEADER
                                        MOV     H.LUN+<<$TMLUN-1>*4>(R1),R$R1+2(SP) ; POINT TO THE NS: UCB
                                        RESMAP                          ;; RESTORE PREVIOUS KISAR6
                                        RETURN                          ;; RETURN TO USER STATE
                                85$:
                                        .IFF    ; DF R$$MPL

001224 017701  000000G                  MOV     @HEADR,R1               ; ELSE, GET OUR HEADER ADDRESS  ; RJK08
001230 016101  000000C                  MOV     H.LUN+<<$TMLUN-1>*4>(R1),R1 ; POINT TO THE NS: UCB

                                        .ENDC   ; DF R$$MPL

001234 132761  000120  000005           BITB    #<US.MNT!US.MDM>,U.STS(R1) ; IS THE NODE ON ?
001242 001070                           BNE     110$                    ; IF NE, NO - ILLEGAL OPERATION
                                ;
                                ; IF ENDNODE THEN CHECK FOR OTHER ACTIVE XPT OWNED CIRCUITS
                                ;
001244 012700  177765                   MOV     #ME.CST,R0              ; Else, assume circuit is in wrong state
001250 012764  000003  000072           MOV     #ME$CIR,L$PAR(R4)       ; ...

001256 032761  000200  000014           BIT     #NF.END,U.CW3(R1)       ; Is this an Endnode ?
001264 001465                           BEQ     130$                    ; If EQ, no

001266 126427  000100  000000           CMPB    L$STA(R4),#MS$ON        ; Else, are we turning state ON ?
001274 001061                           BNE     130$                    ; If NE, no, by-pass further checks

001276 116301  000006                   MOVB    Z.LLN(R3),R1            ; Set up count of channels in list
001302 001450                           BEQ     110$                    ; No channels...hope this never happens!
001304 062703  000022                   ADD     #Z.MAP+2,R3             ; Compute XPT channel list address
001310 005301                           DEC     R1                      ; and skip first list entry

001312                                  SAVRG   R5                      ; Save register
001314 022723  100000           90$:    CMP     #ZS.ASN,(R3)+           ; Is channel free to assign?
001320 001004                           BNE     100$                    ; If NE, then no, check if circuit active
```

```
                                                  .SBTTL  SETCOS - SET COST FOR A CIRCUIT
                                          ;+
                                          ; **SETCON-SET COST FOR A CIRCUIT
                                          ;
                                          ; THIS ROUTINE IS CALLED TO CHANGE THE COST FOR A CIRCUIT.
                                          ; IF THE CIRCUIT IS ON, AND ASSIGNED TO TRANSPORT AN EVENT IS POSTED
                                          ; TO THE ROUTING TASK.
                                          ;
                                          ; INPUTS:
                                          ;       R4 = ADDRESS OF CONTEXT AREA WITH:
                                          ;               L$COS(R4) = NEW COST
                                          ;
                                          ; OUTPUTS:
                                          ;       IF CC, COST CHANGED,
                                          ;       ELSE, ERROR CODE IN R0
                                          ;
                                          ; REGISTERS:
                                          ;       R4, R5 ARE PRESERVED
                                          ;-

002604                                    SETCOS::
                                          ;
                                          ; CHECK THE CIRCUIT STATE
                                          ;
002604  012700  177752                            MOV     #ME.PNA,R0              ; ASSUME INVALID PARAMETER
002610  012764  001604  000072                    MOV     #MP$COS,L$PAR(R4)       ; ...
002616  022764  177777  000050                    CMP     #-1,L$TPT(R4)           ; IS THIS A PVC?
002624  001540                                    BEQ     90$                     ; BR IF YES
002626  012700  177767                            MOV     #ME.IID,R0              ; ASSUME INVALID CIRCUIT-ID
002632  012764  000003  000072                    MOV     #ME$CIR,L$PAR(R4)       ; ...
002640  132764  000010  000041                    BITB    #LP$MPT,L$PFG(R4)       ; IS THIS A MULTI-POINT CIRCUIT ?
002646  001404                                    BEQ     10$                     ; IF EQ, NO
002650  132764  000002  000041                    BITB    #LP$TRB,L$PFG(R4)       ; ELSE, WAS A TRIB NUMBER GIVEN ?
002656  001523                                    BEQ     90$                     ; IF EQ, NO - INVALID CIRCUIT-ID
002660  016401  000042                    10$:    MOV     L$SLT(R4),R1            ; GET THE SLT ADDRESS
002664  012700  177765                            MOV     #ME.CST,R0              ; ASSUME CIRCUIT IN WRONG STATE
002670  032761  040000  000000                    BIT     #LF.RDY,L.FLG(R1)       ; IS THE CIRCUIT LOADED ?
002676  001513                                    BEQ     90$                     ; IF EQ, NO - WRONG STATE
002700  012700  000001                            MOV     #MS.SUC,R0              ; ASSUME SUCCESS
002704  005064  000072                            CLR     L$PAR(R4)               ; NO ERROR DETAIL
002710  105761  000014                            TSTB    L.NSTA(R1)              ; IS THIS A MULTIPOINT MASTER ?
002714  001010                                    BNE     20$                     ; IF NE, YES

002716  116461  000101  000015                    MOVB    L$COS(R4),L.COST(R1)    ; ELSE, SET THE NEW COST
002724  032761  100000  000000                    BIT     #LF.ACT,L.FLG(R1)       ; IS THE CIRCUIT ACTIVE ?
002732  001476                                    BEQ     100$                    ; IF EQ, NO - BYPASS TASK ACTIVATION
002734  000411                                    BR      30$                     ; AND CHECK IF CIRCUIT IS OWNED BY XPT

002736  016402  000050                    20$:    MOV     L$TPT(R4),R2            ; ELSE POINT TO THE TRIB FLAGS
002742  116462  000101  000001                    MOVB    L$COS(R4),S.COST(R2)    ; SET THE NEW COST
002750  132762  000200  000000                    BITB    #SF.ACT,S.FLG(R2)       ; IS THIS TRIBUTARY ACTIVE ?
002756  001464                                    BEQ     100$                    ; IF EQ, NO - BYPASS TASK ACTIVATION
                                          ;
                                          ; SEE IF THE CIRCUIT IS OWNED BY TRANSPORT
                                          ;
002760  017701  000000G                   30$:    MOV     @DECPT,R1               ; POINT TO DEC HOME BLOCK       ; RJK08
002764  005761  000060                            TST     D$NN(R1)                ; IS THIS A NON-ROUTING NODE ?
```

```
                                                                   N  7
CICHPD - Circuit Change Paramet MACRO V05.03b Saturday 29-Jun-85 12:13  Page 24-7
Symbol table

U.OWN    177776          U.TTYP  000046          U2.L8S= 010000          W.BLPD  000016          ZF.X3P= 000000
U.RBNS=  000112          U.TUX   000024          U2.NEC= 004000          W.BLVR  000002          ZS.ASN= 100000
U.RCTC=  000113          U.UHVR= 000107          U2.PRV= 000010          W.BNPD  000015          ZS.BSY= 140000
U.RCTS=  000110          U.UIC   000044          U2.RMT= 020000          W.BOFF  000012          Z.AVL   000014
U.RED    000002          U.UNFL= 000052          U2.R04= 100000          W.BPCB  000000          Z.DAT   000016
U.RED2   000034          U.UNIT  000006          U2.SCS= 000004          W.BSIZ  000010          Z.DSP   000000
U.SCB    000020          U.UNSZ= 000114          U2.SLV= 000200          XMTTBL  000144R         Z.FLG   000010
U.SHST=  000076          U.UNTI= 000060          U2.VT5= 000002          X$$DBT= 000000          Z.LEN = 000016
U.SHUN=  000074          U.USVR= 000106          U2.7CH= 010000          ZF.COU= 001000          Z.LLN   000006
U.SPC =  000036          U.UTIL= 000036          U3.OPA= 100000          ZF.DDM= 000001          Z.MAP   000020
U.STS    000005          U.VCB = 000034          U3.PAR= 040000          ZF.DIA= 004000          Z.NAM   000004
U.ST2    000007          U.VSER= 000120          U3.UPC= 020000          ZF.DLC= 000002          Z.PCB   000012
U.SUB =  000036          U.2MED= 000070          U4.CR = 000100          ZF.DVP= 100000          Z.SCH   000007
U.TCHP   000042          U2.AT.= 000020          VF$RCV= 100000          ZF.INI= 040000          $DMPST= ****** GX
U.TCVP   000043          U2.CRT= 002000          VF$XMT= 040000          ZF.KMX= 000020          $HEADR= ****** GX
U.TFLK   000040          U2.DH1= 100000          V$FLG   000000          ZF.LLC= 000004          $TMLUN= ****** GX
U.TFRQ   000036          U2.DJ1= 040000          V$LEN   000022          ZF.LMC= 000100          $$LCNT= 000154
U.TIXL   000060          U2.DZ1= 000100          V$RCV   000002          ZF.MAN= 020000          $$LLIN= 000076
U.TLPP   000053          U2.ESC= 001000          V$XMT   000012          ZF.MFL= 000010          $$$    = 000062
U.TMTI   000047          U2.HFF= 010000          V$$CTR= 001000          ZF.MTM= 000400          $$$L   = 000023
U.TRCK=  000100          U2.HLD= 000040          W.BATT  000006          ZF.MUX= 000040          $$$SYS= 004374
U.TSTA   000026          U2.LOG= 000400          W.BFPD  000014          ZF.PSE= 002000          $$$SYX= 000000
U.TST5   000054          U2.LWC= 000001          W.BHVR  000004          ZF.SLI= 010000          .$$$$.= 000034
U.TST6   000056          U2.L3S= 000004          W.BLGH  000020          ZF.TIM= 000200          ..OFF.= 177732
U.TTAB   000050

. ABS.   177776    000   (RW,I,GBL,ABS,OVR)
         003132    001   (RW,I,LCL,REL,CON)
$$$UNM   000714    002   (RO,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 367
Work  file writes: 286
Size of work file: 54705 Words  ( 214 Pages)
Size of core pool: 17608 Words  ( 67 Pages)
Operating  system: RSX-11M/PLUS

Elapsed time: 00:01:15.57
SY:CICHPD.V2,[135,134]CICHPD/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[130,10]RSXMCM/PA:1,[135,10]CICHPD
```

```
        .IF NDF R$$11S
        .TITLE  CICHPX - Circuit Change Parameters Operate (X.25 subroutines)
        .IDENT  /V05.00/
        .NLIST  BEX
;
; COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY
; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
;
;
; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND  COPIED
; ONLY  IN  ACCORDANCE  WITH  THE  TERMS  OF  SUCH  LICENSE AND WITH THE
; INCLUSION OF THE ABOVE COPYRIGHT NOTICE.  THIS SOFTWARE OR  ANY  OTHER
; COPIES  THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
; OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE  IS  HEREBY
; TRANSFERRED.
;
; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE  WITHOUT  NOTICE
; AND  SHOULD  NOT  BE  CONSTRUED  AS  A COMMITMENT BY DIGITAL EQUIPMENT
; CORPORATION.
;
; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF  ITS
; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
;
;
; MODULE DESCRIPTION:
;
; NETWORK MANAGEMENT - CIRCUIT CHANGE PARAMETERS OPERATE ROUTINE (X.25)
;
;
; DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
;
; IDENT HISTORY:
;
; 1.00  8-May-83
;       [Tim Dixon EDSSEG]
;       Split from CICHOP (see history in CIHOP) to permit heavier overlaying
;       DLM management functions added.
;       Fix bug in COUNTER TIMER routine for channels > 127
;
; 4.00  07-NOV-83
;       DECNET-11M V4.0
;       DECNET-11M-PLUS V2.0
;
; 5.00  22-JUL-85
;       DECnet-11M/S V4.2
;       DECnet-11M-Plus V3.0
;       DECnet-Micro/RSX V1.0
;
;
```

```
                                         .SBTTL  CKDSVC - Check for DLM SVC
                                 ;+
                                 ; *** CKDSVC - Check for DLM SVC
                                 ;
                                 ; INPUTS
                                 ;       R4 - Address of context area
                                 ;               L$SLT(R4) - Address of SLT entry for disguised line
                                 ;
                                 ; OUTPUTS
                                 ;       Carry clear if circuit is a DLM SVC
                                 ;
                                 ;-

001012                           CKDSVC:
001012                                   SWSTK$  30$                     ;; Enter system state
001016                                   SAVMAP                          ;; Save mapping
001022                                   CALL    FNDCTR                  ;; Find counter block for line
001026  103410                           BCS     10$                     ;; Can't be DLM if no counters
001030  132761  000200  000044           BITB    #PX$DLM,T$FLAG(R1)      ;; Check for DLM circuit
001036  001404                           BEQ     10$                     ;; Flag not set if not DLM
001040  132761  000100  000044           BITB    #PX$SVC,T$FLAG(R1)      ;; If DLM, ensure this is an SVC
001046  001005                           BNE     20$                     ;; Correct if SVC flag set
001050                           10$:    RETC    R0                      ;; Indicate circuit not DLM SVC
001062                           20$:    RESMAP                          ;; Restore mapping
001066                           30$:    RETURN
```

```
                                                                  N 10
CICHPX     CREATED BY  MACRO  ON 29-JUN-85 AT 12:15      PAGE 6
SYMBOL CROSS REFERENCE                                   CREF    04.00

SYMBOL   VALUE            REFERENCES

S$LFT     000012          7-126       7-127
S$PKSZ    000004          7-123       7-124
S$RCL     000013          7-127       7-128
S$TIM     000014          7-128       7-129
S$TMR     000016          7-129       7-130
S$WND     000006          7-124       7-125
S.COST    000001         #5-72
S.FLG     000000         #5-72        11-281
S.LEN     000004         #5-72
S.NMST    000002         #5-72        11-283
S.OWNR    000003         #5-72
T$FLAG    000044         #5-67       *12-315     *12-318     14-365     14-367
T$LIF     000013         #5-67
T$LIFL    000013         #5-67
T$LIFO    000013         #5-67
T$LIFS    000013         #5-67
T$LIN     000000         #5-67
T$LIPS    000006         #5-67
T$LLD     000012         #5-67
T$LLDC    000045         #5-67
T$LLDL    000012         #5-67
T$LLDO    000012         #5-67
T$LLDS    000012         #5-67
T$LLEN    000046         #5-67
T$LOPR    000002         #5-67
T$LTCL    000024         #5-67
T$LTIM    000026         #5-67
T$LTPR    000014         #5-67
T$LTPS    000020         #5-67
T$NAPL    000004         #5-67
T$NFE     000000         #5-67
T$NLEN    000010         #5-67
T$NNUL    000002         #5-67
T$NOPL    000006         #5-67
T$NRNI    000042         #5-67
T$NRPL    000005         #5-67
T$NRUL    000007         #5-67
T$NVR     000001         #5-67
T$RPRI    000040         #5-67
T$SVC     000034         #5-67        10-226      10-227
T$T5      000030         #5-67
T$T6      000032         #5-67
X$CTIM    000060         *15-402
$ADPVC    001222 RG      #16-418
$CEACC  = ******   GX     15-392      15-400      15-401
$HEADR  = ******   GX     14-369
$STPAR  = ******   GX     10-231
$XLBOV  = 000000         #5-89
```

```
    331 001052  001403                          BEQ     90$                     ; BR if no
    332 001054  052764  040000  000064          BIS     #LF$MLT,L$FLG(R4)       ; Else set multiple adjacency flag
    333                                                                         ;     for next time
    334 001062                          90$:    RETURN
```

```
                                                       .SBTTL  SETSER - Set Service parameter
                                              ;+
                                              ; ** SETSER - Set service parameter
                                              ;
                                              ; Inputs:         R0      = SLT pointer
                                              ;                 R1      = Tribs pointer
                                              ;                 R3      = buffer pointer
                                              ;
                                              ; Output:         Service parameter in message. R3 updated.
                                              ;-

 003234                                       SETSER:
 003234  026427  000000  015355                        CMP     L$DDM(R4),#^RDLM        ; DLM circuit ?
 003242  001425                                        BEQ     30$                     ; if EQ, parameter not applicable
 003244  004567  000000G                               JSR     R5,SETPAR               ; Service parameter
 003250  000144                                        .WORD   MP$SER                  ; ...
 003252  000201                                        .WORD   MT$COD!1                ; one byte coded
 003254  112723  000000                                MOVB    #MB$ENA,(R3)+           ; assume enabled
 003260  105760  000014                                TSTB    L.NSTA(R0)              ; multipoint controller or Ethernet ?
 003264  001005                                        BNE     10$                     ; if NE, yes
 003266  032760  000040  000000                        BIT     #LF.SER,L.FLG(R0)       ; is Service disabled ?
 003274  001005                                        BNE     20$                     ; if NE, yes
 003276  000407                                        BR      30$                     ; else done

 003300  132761  000001  000000       10$:             BITB    #SF.SER,S.FLG(R1)       ; is service disabled for circuit ?
 003306  001403                                        BEQ     30$                     ; if EQ, no

 003310  112763  000001  177777       20$:             MOVB    #MB$DIS,-1(R3)          ; else set parameter to disabled value

 003316                               30$:             RETURN
```

```
CIRECH      CREATED BY  MACRO  ON 29-JUN-85 AT 12:16      PAGE 6
SYMBOL CROSS REFERENCE                                     CREF   04.00

SYMBOL  VALUE            REFERENCES

R.LST     000104          #6-74
R.MAP     000110          #6-74
R.QUE     000062          #6-74
R.SLN     000070          #6-74
R.SRV     000060          #6-74
R.STA     000066          #6-74
R.STBL    000210          #6-74
SETNAM    003320 R         16-699     #21-937
SETNOD    002266 R          9-157       9-177     #16-661
SETPAR  = ******   GX       9-182       9-200       9-204       9-210       9-218       9-231       9-249       9-271       9-280
                           11-401      11-414      18-821      18-825      19-862      19-871      19-879      20-902
SETSER    003234 R          9-151     #20-899
SF$INC  = 000002           17-787      18-830
SF.ACT  = 000200          #6-70
SF.ENA  = 000100          #6-70
SF.LPB  = 000004          #6-70
SF.MFL  = 000040          #6-70
SF.PAC  = 000020          #6-70
SF.REA  = 000010          #6-70
SF.SER  = 000001          #6-70        20-912
SF.SVC  = 000002          #6-70
SF.UNL  = 000040          #6-70
SLTMA   = ******   GX      17-743
SVCADD    000006 R        #8-92      *17-779      17-780     *17-782     *17-782      17-786      18-820      19-856
SVCBLK    003114 R          9-307    #19-853
SVCMAP    000004 R        #8-91      *17-778      17-785      18-819      19-855
SVCRCL    003006 R          9-237    #18-817
S$ADSZ    000007           19-860
S$DTE     000020           19-866
S$FLG     000010           17-787      18-830
S$PKSZ    000004           19-874      19-875
S$RCL     000013           18-824
S$TMR     000016           18-828      18-829
S$WND     000006           19-882
S$$BAS  = ******            9-318      10-369
S.COST    000001          #6-70         9-189
S.CTL     000006          #6-74
S.ERR     000010          #6-74
S.FLG     000000          #6-70        20-912
S.LBE     000044          #6-74
S.LEN     000004          #6-70
S.LGTH  = 000050          #6-74
S.LIN     000004          #6-74
S.LINK    177776          #6-74         6-74
S.LSA     000005          #6-74
S.NMST    000002          #6-70         9-246
S.OWNR    000003          #6-70        17-737
S.PLA     000010           11-417
S.PSA     000003          #6-74        11-392
S.RBE     000042          #6-74
S.RCB     000020          #6-74
S.RCV     000014          #6-74
```

```
                                        .SBTTL  DLC LINE COUNTERS DISPATCH TABLE
                                ;
                                ; DISPATCH TABLE FOR LINE COUNTERS BASED ON DLC
                                ; THIS TABLE CAN BE ADDED TO BY A USER DEFINING AN ENTRY
                                ; IN THE PSECT $$DLREA. THE ENTRIES CONTIST OF RAD50 DLC
                                ; PROCESS NAME FOLLOWED BY THE ADDRESS OF THE LINE COUNTERS
                                ; ROUTINE.
                                ;
                                        .IF NDF R$$11S                  ; THESE PSECTS ARE DEFINED IN THE
                                                                        ; NIXE MODULE FOR 11S NICE

000000                                  .PSECT  $$DLCO                  ; DATA LINK COUNTERS TABLE
000000                          DLCTBL:
                                        .IF NDF S$$BAS & R$$RTR

000000  014610  000000G                 .WORD   ^RDCP,DCPCIR            ; READ CIRCUIT COUNTERS ENTRYPOINT

                                          .IF NDF R$$PRO                ; PRO/DECnet

000004  014450  000000G                 .WORD   ^RDA,DACIR              ; READ CIRCUIT COUNTERS ENTRYPOINT
000010  015430  000000G                 .WORD   ^RDMP,DMPCIR            ; FOR DMP DLC COUNTERS
000014  015436  000000G                 .WORD   ^RDMV,DMPCIR            ; FOR DMV DLC COUNTERS
000020  015355  000000G                 .WORD   ^RDLM,DLMCOU            ; READ CIRCUIT COUNTERS ENTRYPOINT
000024  062204  000000G                 .WORD   ^RPCL,PCLCIR            ; READ CIRCUIT COUNTERS ENTRYPOINT

                                          .ENDC ; NDF R$$PRO


                                        .ENDC
000000                                  .PSECT  $$DLC1                  ; END OF TABLE
000000  000000                          .WORD   0

                                        .ENDC

000002                                  .PSECT
```

```
                                                                             N 15
CIREIN - CIRCUIT READ INFORMATI MACRO V05.03b Saturday 29-Jun-85 12:17
Table of contents
```

```
     5-   55     MACRO CALLS AND LOCAL DATA
     5-   62     BIT DEFINITIONS
     6-   68     CIREIN - CIRCUIT READ INFORMATION INITIALIZE
```

B	1	
C	1	
D	1	
E	1	
F	1	
G	1	
H	1	
I	1	
J	1	
K	1	
L	1	
M	1	
N	1	
B	2	
C	2	
D	2	
E	2	
F	2	CICHOP
G	2	
H	2	
I	2	
J	2	
K	2	
L	2	
M	2	
N	2	
B	3	
C	3	
D	3	
E	3	
F	3	
G	3	
H	3	
I	3	
J	3	
K	3	
L	3	
M	3	
N	3	
B	4	
C	4	
D	4	
E	4	
F	4	
G	4	
H	4	
I	4	
J	4	
K	4	CICHPD
L	4	
M	4	
N	4	
B	5	
C	5	
D	5	
E	5	
F	5	
G	5	
H	5	
I	5	
J	5	
K	5	
L	5	
M	5	
N	5	
B	6	
C	6	
D	6	
E	6	
F	6	
G	6	
H	6	
I	6	
J	6	
K	6	
L	6	
M	6	
N	6	
B	7	
C	7	
D	7	
E	7	
F	7	
G	7	
H	7	
I	7	
J	7	
K	7	
L	7	
M	7	
N	7	
B	8	
C	8	
D	8	
E	8	
F	8	
G	8	
H	8	
I	8	
J	8	
K	8	
L	8	CICHPX
M	8	
N	8	
B	9	
C	9	
D	9	
E	9	
F	9	
G	9	
H	9	
I	9	
J	9	
K	9	
L	9	
M	9	
N	9	
B	10	
C	10	
D	10	
E	10	
F	10	
G	10	
H	10	
I	10	
J	10	
K	10	
L	10	
M	10	
N	10	
B	11	
C	11	CIRECH
D	11	
E	11	
F	11	
G	11	
H	11	
I	11	
J	11	
K	11	
L	11	
M	11	
N	11	
B	12	
C	12	
D	12	
E	12	
F	12	
G	12	
H	12	
I	12	
J	12	
K	12	
L	12	
M	12	
N	12	
B	13	
C	13	
D	13	
E	13	
F	13	
G	13	
H	13	
I	13	
J	13	
K	13	
L	13	
M	13	
N	13	
B	14	
C	14	
D	14	
E	14	
F	14	CIRECO
G	14	
H	14	
I	14	
J	14	
K	14	
L	14	
M	14	
N	14	
B	15	
C	15	
D	15	
E	15	
F	15	
G	15	
H	15	
I	15	
J	15	
K	15	
L	15	
M	15	CIREIN
N	15	
B	16	
C	16	
D	16	
E	16	
F	16	
G	16	
H	16	
I	16	
J	16	
K	16	
L	16	
M	16	